Analysis of Nyström method with sequential ridge leverage score sampling

Daniele Calandriello, Alessandro Lazaric, Michal Valko

Analysis of Nyström Method with Sequential Ridge Leverage Score Sampling

Daniele Calandriello
SequeL team
INRIA Lille - Nord Europe

Alessandro Lazaric
SequeL team
INRIA Lille - Nord Europe

Michal Valko
SequeL team
INRIA Lille - Nord Europe

Abstract

Large-scale kernel ridge regression (KRR) is limited by the need to store a large kernel matrix $\mathbf{K}_t$. To avoid storing the entire matrix $\mathbf{K}_t$, Nyström methods subsample a subset of columns of the kernel matrix, and efficiently find an approximate KRR solution on the reconstructed $\widetilde{\mathbf{K}}_t$. The chosen subsampling distribution in turn affects the statistical and computational tradeoffs. For KRR problems, [16, 1] show that a sampling distribution proportional to the *ridge leverage scores* (RLSs) provides strong reconstruction guarantees for $\widetilde{\mathbf{K}}_t$. While exact RLSs are as difficult to compute as a KRR solution, we may be able to approximate them well enough. In this paper, we study KRR problems in a sequential setting and introduce the `INK-Estimate` algorithm, that *incrementally* computes the RLSs estimates. `INK-Estimate` maintains a small *sketch* of $\mathbf{K}_t$, that at each step is used to compute an intermediate estimate of the RLSs. First, our sketch update does not require access to previously seen columns, and therefore a *single pass* over the kernel matrix is sufficient. Second, the algorithm requires a fixed, small space budget to run dependent only on the *effective dimension* of the kernel matrix. Finally, our sketch provides strong approximation guarantees on the distance $\|\mathbf{K}_t - \widetilde{\mathbf{K}}_t\|_2$, and on the statistical risk of the approximate KRR solution at *any time*, because all our guarantees hold at any intermediate step.

1 INTRODUCTION

Kernel ridge regression [17, 18] (KRR) is a common nonparametric regression method with well studied theoretical advantages. Its main drawback is that, for n samples, storing and manipulating the kernel regression matrix $\mathbf{K}_n$ requires $\mathcal{O}(n^2)$ space, and can become quickly intractable when n grows. This includes batch large scale KRR, and online KRR, where the size of the dataset t grows over time as new samples are added to the problem. For this purpose, many different methods [23, 4, 10, 14, 11, 24] attempt to reduce the memory required to store the kernel matrix, while still producing an accurate solution.

For the batch case, the Nyström family of algorithms randomly selects a subset of m columns from the kernel matrix $\mathbf{K}_n$ that are used to construct a low rank approximation $\widetilde{\mathbf{K}}_t$ that requires only $\mathcal{O}(nm)$ space to store. The low-rank matrix is then used to find an approximate solution to the KRR problem. The quality of the approximate solution is strongly affected by the sampling distribution and the number of columns selected [16]. For example, uniform sampling is an approach with little computational overhead, but does not work well for datasets with high coherence [7], where the columns are weakly correlated. In particular, Bach [2] shows that the number of columns m necessary for a good approximation when sampling uniformly scales linearly with the maximum degree of freedoms of the kernel matrix. In linear regression, the notion of coherence is strongly related to the definition of leverage points or *leverage scores* of the dataset [6], where points with high (statistical) leverage score are more influential in the regression problem. For KRR, Alaoui and Mahoney [1] introduce a similar concept of *ridge* leverage scores (RLSs) of a square matrix, and shows that Nyström approximations sampled according to RLS have strong reconstruction guarantees of the form $\|\mathbf{K}_n - \widetilde{\mathbf{K}}_n\|_2$, that translate into good guarantees for the approximate KRR solution [1, 16]. Compared to the uniform distribution, a distribution based on RLSs better captures non-uniformities in the data, and can achieve good approximations using only a number of columns m, proportional to the average degrees of freedom of the matrix, called the *effective dimension* of the problem. The disadvantage of RLSs compared to uniform sampling is the high computational cost of exact RLSs, which is comparable to solving KRR itself. Alaoui and Mahoney [1] reduces this problem by showing that a distribution based on approximate RLSs can also provide the same strong guarantees, if the RLSs are approximated up to a constant er-

ror factor. They provide a fast method to compute these RLSs, but, unlike our approach, requires multiple passes over data. Another disadvantage of their approach, that we address, is the *inverse dependence on the minimal eigenvalue* of the kernel matrix in the error bound of Alaoui and Mahoney [1], which can be significant.

While Nyström methods are a typical choice in a batch setting, *online kernel sparsification* (OKS) [4, 5] examines each sample in the dataset sequentially. OKS maintains a small *dictionary* of relevant samples. Whenever a new sample arrive, if the dictionary is not able to accurately represent the new sample as a combination of the samples already stored, the dictionary is updated. This dictionary can be used to approximate KRR incrementally. OKS decides whether to include a sample using the correlation between samples in the dictionary and the new sample. This can measured using approximate linear dependency (ALD) [5], coherence [15], or the surprise criterion [12].

Generalization properties of online kernel sparsification were studied by Engel et al. [5], but depend on the empirical error and are not compared with an exact KRR solution on the whole dataset. Online kernel regression with the ALD rule was analyzed by Sun et al. [19], under the assumption that, asymptotically in n, the eigenvalues of the kernel matrix decay exponentially fast. Sun et al. [19] show that in this case the size of the dictionary grows sublinearly in t, or in other words that, asymptotically in n, the dictionary size converge to a fraction of n that will be small whenever the eigenvalues decay fast enough. This space guarantee is weaker than the fixed space requirements of Nyström methods, one of the reasons is that these methods (unlike ours) cannot remove a sample from the dictionary after inclusion. Furthermore, Van Vaerenbergh et al. [22] studies variants of online kernel regression with a forgetting factor for time-varying series, but these methods are not well studied in the normal KRR setting. Unlike in the batch setting, in the sequential setting we often require the guarantees not only at the end but also *in the intermediate steps* and this is our objective. Inspired by the advances in the analyses of the Nyström methods, in this paper, we focus on finding a space efficient algorithm capable of solving KRR problems in the sequential setting but that would be also equipped with generalization guarantees.

Main contributions We propose the `INK-Estimate` algorithm that processes a dataset $\mathcal{D}$ of size n in *a single pass*. It requires only a small, fixed space budget, $\overline{q}$ proportional to the effective dimension of the problem and on the accuracy required. The algorithm maintains a Nyström approximation $\widetilde{\mathbf{K}}_t$, of the kernel matrix at time t, $\mathbf{K}_t$, based on RLSs estimates. At each step, it uses only the approximation and the newly received sample to incrementally update the RLSs estimate, and to compute $\widetilde{\mathbf{K}}_{t+1}$. Unlike in the batch Nyström setting, our challenge is to track RLSs and an effective dimension that *changes over time*. Sampling distributions based on RLSs can become obsolete and biased, but we show how to update them over time *without necessity of accessing previously seen samples* outside of the ones contained in $\mathbf{K}_t$. Our space budget $\overline{q}$ scales with the average degree of freedom of the matrix, and not the larger maximum degree of freedom (as by Bach [2]), and does not imposes assumptions on the ridge regularization parameter, or on the smallest eigenvalue of the problem as the result of Alaoui and Mahoney [1]. However, we provide the same strong guarantees as batch RLSs based Nyström methods on $\|\mathbf{K}_n - \widetilde{\mathbf{K}}_n\|_2$ and on the risk of the approximate KRR solution. In addition to batch Nyström methods, all of these guarantees hold at any intermediate step t, and therefore the algorithm can output *accurate intermediate solutions*, or it can be interrupted at *any time* and return a solution with guarantees. Finally, it operates in a *sequential* setting, requiring only a single pass over the data.

If we compare `INK-Estimate` to other online kernel regression methods (such as OKS), our algorithm provides generalization guarantees with respect to the exact KRR solution. Furthermore, it provides a new criteria for inclusion of a sample in the dictionary, in particular the ridge leverage scores. This criterion gives us a procedure that not only randomly includes samples in the dictionary, but that also randomly discards them to satisfy space constraints not only asymptotically, but at every step.

2 BACKGROUND

In this section we introduce the notation used through the paper and we introduce the kernel ridge regression problem [17] and Nyström approximation of the kernel matrix with ridge leverage scores.

Notation. We use curly capital letters $\mathcal{A}$ for collections. We use upper-case bold letters $\mathbf{A}$ for matrices, lower-case bold letters $\mathbf{a}$ for vectors, and lower-case letters a for scalars. We denote by $[\mathbf{A}]_{ij}$ and $[\mathbf{a}]_i$ the (i,j) element of a matrix and ith element of a vector respectively. We denote by $\mathbf{I}_n \in \mathbb{R}^{n\times n}$ the identity matrix of dimension n and by $\mathrm{Diag}(\mathbf{a}) \in \mathbb{R}^{n\times n}$ the diagonal matrix with the vector $\mathbf{a} \in \mathbb{R}^n$ on the diagonal. We use $\mathbf{e}_{i,n} \in \mathbb{R}^n$ to denote the indicator vector for element i of dimension n. When the dimensionality of $\mathbf{I}$ and $\mathbf{e}_i$ is clear from the context, we omit the n. We use $\mathbf{A} \succeq \mathbf{B}$ to indicate that $\mathbf{A} - \mathbf{B}$ is a PSD matrix. Finally, the set of integers between 1 and n is denoted by $[n] := \{1, \dots, n\}$.

2.1 Exact Kernel Ridge Regression

Kernel regression. We consider a regression dataset $\mathcal{D} = \{(\mathbf{x}_t, y_t)\}_{t=1}^n$, with input $\mathbf{x}_t \in \mathcal{X} \subseteq \mathbb{R}^d$ and output $y_t \in \mathbb{R}$. We denote by $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ a positive definite kernel function and by $\varphi : \mathcal{X} \to \mathbb{R}^D$ the corresponding

feature map,[1] so that the kernel is obtained as $\mathcal{K}(\mathbf{x}, \mathbf{x}') = \varphi(\mathbf{x})^\mathsf{T}\varphi(\mathbf{x}')$. Given the dataset $\mathcal{D}$, we define the kernel matrix $\mathbf{K}_t \in \mathbb{R}^{t\times t}$ constructed on the first t samples as the application of the kernel function on all pairs of input values, i.e., $[\mathbf{K}_t]_{ij} = \mathcal{K}(\mathbf{x}_i, \mathbf{x}_j)$ for any $i, j \in [t]$ and we denote by $\mathbf{y}_t \in \mathbb{R}^t$ the vector with components y_i, $i \in [t]$. We also define the feature vectors $\boldsymbol{\phi}_t = \varphi(\mathbf{x}_t) \in \mathbb{R}^D$ and after introducing the feature matrix

$$\boldsymbol{\Phi}_t = \left[\begin{array}{c|c|c|c} \boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \dots & \boldsymbol{\phi}_t \end{array}\right] \in \mathbb{R}^{D\times t},$$

we can rewrite the kernel matrix as $\mathbf{K}_t = \boldsymbol{\Phi}_t^\mathsf{T}\boldsymbol{\Phi}_t$. Whenever a new point $\mathbf{x}_{t+1}$ arrives, the kernel matrix $\mathbf{K}_{t+1} \in \mathbb{R}^{t+1\times t+1}$ is obtained by bordering $\mathbf{K}_t$ as

$$\mathbf{K}_{t+1} = \left[\begin{array}{c|c} \mathbf{K}_t & \bar{\mathbf{k}}_{t+1} \\ \hline \bar{\mathbf{k}}_{t+1}^\mathsf{T} & k_{t+1} \end{array}\right] \tag{1}$$

where $\bar{\mathbf{k}}_{t+1} \in \mathbb{R}^t$ is such that $[\bar{\mathbf{k}}_{t+1}]_i = \mathcal{K}(\mathbf{x}_{t+1}, \mathbf{x}_i)$ for any $i \in [t]$ and $k_{t+1} = \mathcal{K}(\mathbf{x}_{t+1}, \mathbf{x}_{t+1})$. According to the definition of the feature matrix $\boldsymbol{\Phi}_t$, we also have $\mathbf{k}_{t+1} = \boldsymbol{\Phi}_t^\mathsf{T}\boldsymbol{\phi}_{t+1}$.

At any time t, the objective of *sequential* kernel regression is to find the vector $\widehat{\mathbf{w}}_t \in \mathbb{R}^t$ that minimizes the regularized quadratic loss

$$\widehat{\mathbf{w}}_t = \arg\min_{\mathbf{w}} \|\mathbf{y}_t - \mathbf{K}_t\mathbf{w}\|^2 + \mu\|\mathbf{w}\|^2, \tag{2}$$

where $\mu \in \mathbb{R}$ is a regularization parameter. This objective admits the closed form solution

$$\widehat{\mathbf{w}}_t = (\mathbf{K}_t + \mu\mathbf{I})^{-1}\mathbf{y}_t. \tag{3}$$

In the following, we use $\mathbf{K}_t^\mu$ as a short-hand for $(\mathbf{K}_t + \mu\mathbf{I})$. In batch regression, $\widehat{\mathbf{w}}_n$ is computed only once when all the samples of $\mathcal{D}$ are available, solving the linear system in Eq. 3 with $\mathbf{K}_n$. In the *fixed-design* kernel regression, the accuracy of resulting solution $\widehat{\mathbf{w}}_n$ is measured by the prediction error on the input set from $\mathcal{D}$. More precisely, the prediction of the estimator $\widehat{\mathbf{w}}_n$ in each point is obtained as $[\mathbf{K}_n\widehat{\mathbf{w}}_n]_i$, while the outputs y_i in the dataset are assumed to be a noisy observation of an unknown target function $f^* : \mathcal{X} \to \mathbb{R}$, evaluated in x_i i.e., for any $i \in [n]$,

$$y_i = f^*(x_i) + \eta_i,$$

where η_i is a zero-mean i.i.d. noise with bounded variance σ^2. Let $\mathbf{f}^* \in \mathbb{R}^n$ be the vector with components $f^*(x_i)$, then the risk of $\widehat{\mathbf{w}}_n$ is measured as

$$\mathcal{R}(\widehat{\mathbf{w}}_n) = \mathbb{E}_\eta\left[||\mathbf{f}^* - \mathbf{K}_n\widehat{\mathbf{w}}_n||_2^2\right]. \tag{4}$$

If the regularization parameter μ is properly tuned, it is possible to show that $\widehat{\mathbf{w}}_n$ has near-optimal risk guarantees (in a minmax sense). Nonetheless, the computation of $\widehat{\mathbf{w}}_n$ requires $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ space, which becomes rapidly unfeasible for large datasets.

[1] where D can be very large or infinite (e.g. gaussian kernel)

2.2 Nyström Approximation with Ridge Leverage Scores

A common approach to reduce the complexity of kernel regression is to (randomly) select a subset of m samples out of $\mathcal{D}$, and compute the kernel between two points only when one of them is in the selected subset. This is equivalent to selecting a subset of columns of the $\mathbf{K}_n$ matrix. More formally, given the n samples in $\mathcal{D}$, a probability distribution $\mathbf{p}_n = [p_{1,n}, \dots, p_{n,n}]$ is defined over all columns of $\mathbf{K}_n$ and $m \leq n$ columns are randomly sampled with replacement according to $\mathbf{p}_n$. We define by $\mathcal{I}_n$ the sequence of m indices $i \in [n]$ selected by the sampling procedure. From $\mathcal{I}_n$, we construct the corresponding selection matrix $\mathbf{S}_n \in \mathbb{R}^{n\times m}$, where each column $[\mathbf{S}_n]_{:,t} \in \mathbb{R}^n$ is all-zero except from the entry corresponding to the t-th element in $\mathcal{I}_n$ (i.e., $[\mathbf{S}]_{ij}$ is non-zero if at trial j the element i is selected). Whenever the non-zero entries of $\mathbf{S}_n$ are set to 1, sampling m columns from matrix $\mathbf{K}_n$ is equivalent to computing $\mathbf{K}_n\mathbf{S}_n \in \mathbb{R}^{n\times m}$. More generally, the non-zero entries of $\mathbf{S}_n$ could be set to some arbitrary weight $[\mathbf{S}]_{ij} = b_{ij}$. The resulting regularized Nyström approximation of the original kernel $\mathbf{K}_n$ is defined as

$$\widetilde{\mathbf{K}}_n = \mathbf{K}_n\mathbf{S}_n(\mathbf{S}_n^\mathsf{T}\mathbf{K}_n\mathbf{S}_n + \gamma\mathbf{I}_m)^{-1}\mathbf{S}_n^\mathsf{T}\mathbf{K}_n, \tag{5}$$

where γ is a regularization term (possibly different from μ). At this point, $\widetilde{\mathbf{K}}_n$ can be used to solve Eq. 3. Let $\mathbf{W} = (\mathbf{S}_n^\mathsf{T}\mathbf{K}_n\mathbf{S}_n + \gamma\mathbf{I}_m)^{-1} \in \mathbb{R}^{m\times m}$ and $\mathbf{C} = \mathbf{K}_n\mathbf{S}_n\mathbf{W}^{1/2} \in \mathbb{R}^{n\times m}$, applying the Woodbury inversion formula [8] we have

$$\begin{aligned}\widetilde{\mathbf{w}}_n =& (\widetilde{\mathbf{K}}_n + \mu\mathbf{I}_n)^{-1}\mathbf{y}_n = (\mathbf{C}\mathbf{I}_m\mathbf{C}^\mathsf{T} + \mu\mathbf{I}_n)^{-1}\mathbf{y}_n \\ =& \left(\frac{1}{\mu}\mathbf{I}_n - \frac{1}{\mu^2}\mathbf{I}_n\mathbf{C}\left(\mathbf{I}_m + \frac{1}{\mu}\mathbf{C}^\mathsf{T}\mathbf{C}\right)^{-1}\mathbf{C}^\mathsf{T}\mathbf{I}_n\right)\mathbf{y}_n \\ =& \frac{1}{\mu}\left(\mathbf{y}_n - \mathbf{C}\left(\mathbf{C}^\mathsf{T}\mathbf{C} + \mu\mathbf{I}_m\right)^{-1}\mathbf{C}^\mathsf{T}\mathbf{y}_n\right).\end{aligned} \tag{6}$$

Computing $\mathbf{W}^{1/2}$ and $\mathbf{C}$ takes $\mathcal{O}(m^3)$ and $\mathcal{O}(nm^2)$ time using a singular value decomposition, and so does solving the linear system. All the operations require to store at most an $n \times m$ matrix. Therefore the final complexity is reduced from $\mathcal{O}(n^3)$ to $\mathcal{O}(nm^2 + m^3)$ time, and from $\mathcal{O}(n^2)$ to $\mathcal{O}(nm)$ space. Rudi et al. [16] recently showed that in random design, the risk of the resulting solution $\widetilde{\mathbf{w}}_n$ strongly depends on the choice of m and the column sampling distribution $\mathbf{p}_n$. Early methods sampled columns uniformly, and Bach [2] shows that the using this distribution can provide a good approximation when the maximum diagonal entry of $\mathbf{K}_n(\mathbf{K}_n + \mu\mathbf{I})^{-1}$ is small. Following on this approach, Alaoui and Mahoney [1] propose a distribution proportional to these diagonal entries and calls them γ-Ridge Leverage Scores. We now restate their definition of RLS, corresponding sampling distribution, and the effective dimension.

Definition 1. *Given a kernel matrix $\mathbf{K}_n \in \mathbb{R}^{n\times n}$, the γ-ridge leverage score (RLS) of column $i \in [n]$ is*

$$\tau_{i,n}(\gamma) = \mathbf{k}_{i,n}^{\mathsf{T}}(\mathbf{K}_n + \gamma \mathbf{I}_m)^{-1}\mathbf{e}_{i,n}, \tag{7}$$

where $\mathbf{k}_{i,n} = \mathbf{K}_n\mathbf{e}_{i,n}$. Furthermore, the effective dimension $d_{\mathit{eff}}(\gamma)_n$ of the kernel is defined as

$$d_{\mathit{eff}}(\gamma)_n = \sum_{i=1}^{n} \tau_{i,n}(\gamma) = \mathrm{Tr}\left(\mathbf{K}_n(\mathbf{K}_n + \gamma \mathbf{I}_n)^{-1}\right). \tag{8}$$

The corresponding sampling distribution $\mathbf{p}_n$ is defined as

$$[\mathbf{p}_n]_i = p_{i,n} = \frac{\tau_{i,n}(\gamma)}{\sum_{j=1}^{n}\tau_{i,n}(\gamma)} = \frac{\tau_{i,n}}{d_{\mathit{eff}}(\gamma)_n}. \tag{9}$$

The RLSs are directly related to the structure of the kernel matrix and the regularized regression. If we perform an eigendecomposition of the kernel matrix as $\mathbf{K}_n = \mathbf{U}_n\mathbf{\Lambda}_n\mathbf{U}_n^{\mathsf{T}}$, then the RLS of a column $i \in [n]$ is

$$\tau_{i,n}(\gamma) = \sum_{j=1}^{n} \frac{\lambda_j}{\lambda_j + \gamma}[\mathbf{U}]_{i,j}^2, \tag{10}$$

which shows how the RLS is a weighted version of the standard leverage scores (i.e., $\sum_j [U]_{i,j}^2$), where the weights depend on both the spectrum of $\mathbf{K}_n$ and the regularization γ, which plays the role of a soft threshold on the rank of $\mathbf{K}_n$. Similar to the standard leverage scores [3], the RLSs measure the relevance of each point $\mathbf{x}_i$ for the overall kernel regression problem. Another interesting property of the RLSs is that their sum is the effective dimension $d_{\mathrm{eff}}(\gamma)_n$, which measures the intrinsic capacity of the kernel $\mathbf{K}_n$ when its spectrum is soft-thresholded by a regularization γ.[2] We refer to the overall Nyström method using RLS and sampling according to to $\mathbf{p}_n$ in Eq. 9 as `Batch-Exact`, which is illustrated in Alg. 1. We single out the `Direct-sample` subroutine (which simply draws m independent samples from the multinomial distribution $\mathbf{p}_n$) to ease the introduction of our incremental algorithm in the next section.

With the following claim, Alaoui and Mahoney [1] prove that the regularized Nyström approximation $\widetilde{\mathbf{K}}_n$ obtained from Eq. 5 guarantees an accurate reconstruction of the original kernel matrix $\mathbf{K}_n$, and the risk of the associated solution $\widetilde{\mathbf{w}}_n$ is close to the risk of the exact solution $\widehat{\mathbf{w}}_n$.

Proposition 1 (Alaoui and Mahoney [1], App. A, Lem. 1)**.** *Let $\gamma \geq 1$, let $\mathbf{K}_n$ be the full kernel matrix ($t = n$), and let $\tau_{i,n}$, $d_{\mathit{eff}}(\gamma)_n$, $p_{i,n}$ be defined according to Definition 1. For any $0 \leq \varepsilon \leq 1$, and $0 \leq \delta \leq 1$, if we run Alg. 1 using `Direct-sample` (Subroutine 1) with sampling budget m,*

$$m \geq \left(\frac{2d_{\mathit{eff}}(\gamma)}{\varepsilon^2}\right)\log\left(\frac{n}{\delta}\right),$$

[2] Notice that indeed we have $d_{\mathrm{eff}}(\gamma)_n \leq \mathrm{Rank}(\mathbf{K}_n)$.

Algorithm 1 `Batch-Exact` algorithm

Input: $\mathcal{D}$, regularization parameter γ, sampling budget m and probabilities $\mathbf{p}_n$ (Eq. 9)
Output: Nyström approximation $\widetilde{\mathbf{K}}_n$, matrix $\mathbf{S}_n$

1: Compute $\mathcal{I}_n$ using `Direct-sample`$(\mathbf{p}_n, m)$
2: Compute $\mathbf{S}_n$ using $\mathcal{I}_n$ and weights $1/\sqrt{mp_{i,n}}$
3: Compute $\widetilde{\mathbf{K}}_n$ using $\mathbf{S}_n$ and Eq. 5

Subroutine 1 `Direct-sample`$(\mathbf{p}_n, m) \to \mathcal{I}_n$

Input: probabilities $\mathbf{p}_n$, sampling budget m
Output: subsampled column indices $\mathcal{I}_n$

1: **for** $j = \{1, \ldots, m\}$ **do**
2: Sample $i \sim \mathcal{M}(p_{1,n}, \ldots, p_{n,n})$
3: Add i to $\mathcal{I}_n$
4: **end for**

to compute matrix $\mathbf{S}_n$, then with probability $1-\delta$ the corresponding Nyström approximation $\widetilde{\mathbf{K}}_n$ in Eq. 5 satisfies the condition

$$\mathbf{0} \preceq \mathbf{K}_n - \widetilde{\mathbf{K}}_n \preceq \frac{\gamma}{1-\varepsilon}\mathbf{K}_n(\mathbf{K}_n + \gamma\mathbf{I}_n)^{-1} \preceq \frac{\gamma}{1-\varepsilon}\mathbf{I}_n. \tag{11}$$

Furthermore, replacing $\mathbf{K}_n$ by $\widetilde{\mathbf{K}}_n$ in Eq. 3 gives an approximation solution $\widetilde{\mathbf{w}}_n$ such that

$$\mathcal{R}(\widetilde{\mathbf{w}}_n) \leq \left(1 + \frac{\gamma}{\mu}\frac{1}{1-\varepsilon}\right)^2 \mathcal{R}(\widehat{\mathbf{w}}_n).$$

Discussion This result directly relates the number of columns selected m with the accuracy of the approximation of the kernel matrix. In particular, the inequalities in Eq. 11 show that the distance $\|\mathbf{K}_n - \widetilde{\mathbf{K}}_n\|_2$ is smaller than $\gamma/(1-\varepsilon)$. This level of accuracy is then sufficient to guarantee that, when γ is properly tuned, the prediction error of $\widetilde{\mathbf{w}}_n$ is only a factor $(1+2\varepsilon)^2$ away from the error of the exact solution $\widehat{\mathbf{w}}$. As it was shown in [1], using $\widetilde{\mathbf{K}}_n$ in place of $\mathbf{K}_n$ introduces a bias in the solution $\widetilde{\mathbf{w}}_n$ of order γ. For appropriate choices of γ this bias is dominated by the ridge regularization bias controlled by μ. As a result, $\widetilde{\mathbf{w}}_n$ can indeed achieve almost the same risk as $\widehat{\mathbf{w}}_n$ and, at the same time, ignore all directions that are whitened by the regularization and only approximate those that are more relevant for ridge regression, thus reducing both time and space complexity. The RLSs quantify how important each column is to approximate these relevant directions but computing exact RLSs $\tau_{i,n}(\gamma)$ using Eq. 7 is as hard as solving the regression problem itself. Fortunately, in many cases it is computationally feasible to find an approximation of the RLSs. Alaoui and Mahoney [1] explore this possibility, showing that the accuracy and space guarantees are robust to perturbations in the distribution $\mathbf{p}_n$, and provide a two-pass method to compute such approximations. Unfortunately, the accuracy of their RLSs approximation is proportional to the smallest eigenvalue $\lambda_{\min}(\mathbf{K}_n)$, which in

Algorithm 2 The INK-Oracle algorithm

Input: Dataset $\mathcal{D}$, regularization γ, sampling budget $\overline{q}$ and (α, β)-oracle

Output: $\widetilde{\mathbf{K}}_n, \mathbf{S}_n$

Initialize $\mathcal{I}_0$ as empty, $\widetilde{p}_{1,0} = 1, b_{1,0} = 1$, budget $\overline{q}$
for $t = 0, \dots, n-1$ **do**
  Receive new column $\overline{\mathbf{k}}_{t+1}$ and scalar k_{t+1}
  Receive α-leverage scores $\widetilde{\tau}_{i,t+1}$ for any $i \in \mathcal{I}_t \cup \{t+1\}$ from (α, β)-oracle
  Receive β-approximate $\widetilde{d}_{\text{eff}}(\gamma)_{t+1}$ from (α, β)-oracle
  Set $\widetilde{p}_{i,t+1} = \min\{\widetilde{\tau}_{i,t+1}/\widetilde{d}_{\text{eff}}(\gamma)_{t+1}, \widetilde{p}_{i,t}\}$
  $\mathcal{I}_{t+1}, \mathbf{b}_{t+1}$ = Shrink-Expand$(\mathcal{I}_t, \widetilde{\mathbf{p}}_{t+1}, \mathbf{b}_t, \overline{q})$
  Compute $\mathbf{S}_{t+1}$ using $\mathcal{I}_{t+1}$ and weights $\sqrt{b_{i,t+1}}$
  Compute $\widetilde{\mathbf{K}}_{t+1}$ using $\mathbf{S}_{t+1}$ and Equation 5
end for
Return $\widetilde{\mathbf{K}}_n$ and $\mathbf{S}_n$

some cases can be very small. In the rest of the paper, we propose an *incremental* approach that requires only a *single pass* over the data and, at the same time, does not depend on $\lambda_{\min}(\mathbf{K}_n)$ to be large as in [1], or on $\max_i \tau_{i,n}$ to be small as in [2].

3 INCREMENTAL ORACLE KERNEL APPROXIMATION WITH SEQUENTIAL SAMPLING

Our main goal is to extend the known ridge leverage score sampling to the *sequential setting*. This comes with several challenges that needs to be addressed *simultaneously:*

1. The RLSs change when a new sample arrives. We not only need to estimate them, but to *update* this estimate over iterations.

2. The effective dimension $\widetilde{d}_{\text{eff}}(\gamma)_t$, necessary to normalize the leverage scores for the sampling distribution $\mathbf{p}_n$, depends on the interactions of all columns, including the ones that we decided *not* to keep.

3. Due to changes in RLSs, our sampling distribution $\widetilde{\mathbf{p}}_t$ *changes over time*. We need to update to dictionary to reflect these changes, or it will quickly become *biased*, but once we completely drop a column, we cannot sample it again.

In this section, we address the third challenge of incremental updates of the columns with an algorithm for the approximation of the kernel matrix $\mathbf{K}_n$, assuming that the first and second issue are addressed by an *oracle* giving

Subroutine 2 Shrink-Expand$(\mathcal{I}_t, \widetilde{\mathbf{p}}_{t+1}, \mathbf{b}_t, \overline{q})$

Input: $\mathcal{I}_t$, app. pr. $\{(\widetilde{p}_{i,t+1}, b_{i,t}) : i \in \mathcal{I}_t\}$, $\widetilde{p}_{t+1,t+1}$, $\overline{q}$
Output: $\mathcal{I}_{t+1}$

for all $i \in \{1, \dots, t\}$ **do** ▷Shrink
  $b_{i,t+1} = b_{i,t}$
  while $b_{i,t+1}\widetilde{p}_{i,t+1} \leq 1/\overline{q}$ and $b_{i,t} \neq 0$ **do**
    Sample a random Bernoulli $\mathcal{B}\left(\frac{b_{i,t+1}}{b_{i,t+1}+1}\right)$
    On success set $b_{i,t+1} = b_{i,t+1} + 1$
    On failure set $b_{i,t+1} = 0$
  end while
end for
$b_{t+1,t+1} = 1$ ▷Expand
while $b_{t+1,t+1}\widetilde{p}_{t+1,t+1} \leq 1/\overline{q}$ and $b_{t+1,t+1} \neq 0$ **do**
  Sample a random Bernoulli $\mathcal{B}\left(\frac{b_{t+1,t+1}}{b_{t+1,t+1}+1}\right)$
  On success set $b_{t+1,t+1} = b_{t+1,t+1} + 1$
  On failure set $b_{t+1,t+1} = 0$
end while
Add to $\mathcal{I}_{t+1}$ all columns with $b_{i,t+1} \neq 0$

both good approximations of leverage scores and the effective dimension.

Definition 2. *At any step t, an (α, β)-oracle returns an α-approximate ridge leverage scores $\widetilde{\tau}_{i,t}$ which satisfy*

$$\frac{1}{\alpha}\tau_{i,t}(\gamma) \leq \widetilde{\tau}_{i,t} \leq \tau_{i,t}(\gamma),$$

for any $i \in [t]$ and and a β-approximate effective dimension $\widetilde{d}_{\text{eff}}(\gamma)_t$ which satisfy

$$d_{\text{eff}}(\gamma)_t \leq \widetilde{d}_{\text{eff}}(\gamma)_t \leq \beta d_{\text{eff}}(\gamma)_t.$$

We address the first and second challenge in Sect. 4 with an efficient implementation and (α, β)-oracle. In the following we give the *incremental* INK-Oracle algorithm equipped with an (α, β)-oracle that after n steps it returns a kernel approximation with the same properties as if an (α, β)-oracle was used directly at time n.

3.1 The INK-Oracle Algorithm

Apart from an (α, β)-oracle and the dataset $\mathcal{D}$, INK-Oracle (Alg. 2) receives as input the regularization parameter γ used in constructing the final Nyström approximation and a sampling budget $\overline{q}$. It initializes the index dictionary $\mathcal{I}_0$ of stored columns as empty, and the estimated probabilities as $\widetilde{p}_{i,0} = 1$. Finally it initializes a set of integer weights $b_{i,0} = 1$. These weights will represent a discretized approximation of $1/\widetilde{p}_{i,t}$ (the inverse of the probabilities). At each time step t, it receives a new column $\overline{\mathbf{k}}_{t+1}$ and k_{t+1}. This can be implemented either by having a separate algorithm, constructing each column sequentially and

stream it to INK-Oracle, or by having INK-Oracle store just the samples (for an additional $\mathcal{O}(td)$ space complexity) and independently compute the column once. The algorithm invokes the (α, β)-oracle to compute approximate probabilities $\widetilde{p}_{i,t+1} = \widetilde{\tau}_{i,t+1}/\widetilde{d}_{\text{eff}}(\gamma)_{t+1}$, and then takes the minimum $\min\{\widetilde{p}_{i,t+1}, \widetilde{p}_{i,t}\}$ for the sampling probability. As our analysis will reveal, this step is necessary to ensure that the Shrink-Expand operation remains well defined, since the true probabilities $p_{i,t}$ decrease over time. It is important to notice that differently from the batch sampling setting, the approximate probabilities do not necessarily sum to one, but it is guaranteed that $\sum_{i=1}^{t} \widetilde{p}_{i,t} \leq 1$. The Shrink-Expand procedure is composed of two steps. In the Shrink step, we update the weights of the columns already in our dictionary. To decide whether a weight should be increased or not, the product of the weight at the preceding step $b_{i,t-1}$ and the new estimate $\widetilde{p}_{i,t}$ is compared to a threshold. If the product is above the threshold, it means the probability did not change much, and no action is necessary. If the product falls below the threshold, it means the decrease of $\widetilde{p}_{i,t}$ is significant, and the old weight is not representative anymore and should be increased. To increase the weight (e.g. from k to $k+1$), we draw a Bernoulli random variable $\mathcal{B}(\frac{k}{k+1})$, and if it succeeds we increase the weight to $k+1$, while if it fails we set the weight to 0. The more $\widetilde{p}_{i,t}$ decrease over time, the higher the chanches that $b_{i,t+1}$ is set to zero, and the index i (and the associated column $\mathbf{k}_{i,t+1}$) is completely dropped from the dictionary. Therefore, the Shrink step randomly reduces the size of the dictionary to reflect the evolution of the probabilities. Conversely, the Expand step introduces the new column in the dictionary, and quickly updates its weight $b_{t,t}$ to reflect $\widetilde{p}_{t,t}$. Depending on the relevance (encoded by the RLS) of the new column, this means that it is possible that the new column is discarded at the same iteration as it is introduced. For a whole pass over the dataset, INK-Oracle queries the oracle for each RLS at least once, but it *never* asks again for the RLS of a columns dropped from $\mathcal{I}_t$. As we will see in the next section, this greatly simplifies the construction of the oracle. Finally, after updating the dictionary, we use the updated weights $\sqrt{b_{i,t}}$ to update the approximation $\widetilde{\mathbf{K}}_t$, that can be used at any time and not only in the end.

3.2 Analysis of INK-Oracle

The main result of this section is the lower bound on the number of columns required to be kept in order to guarantee a $\gamma/(1-\varepsilon)$ approximation of $\mathbf{K}_t$.

Theorem 1. *Let $\gamma > 1$. Given access to an (α, β)-oracle, for $0 \leq \varepsilon \leq 1$ and $0 \leq \delta \leq 1$, if we run Alg. 2 with parameter $\overline{q}$*

$$\overline{q} \geq \left(\frac{28\alpha\beta d_{\mathit{eff}}(\gamma)_t}{\varepsilon^2}\right) \log\left(\frac{4t}{\delta}\right),$$

to compute a sequence of random matrices $\mathbf{S}_t$ with a random number of columns Q_t, then with probability $1-\delta$, for all t the corresponding Nyström approximation $\widetilde{\mathbf{K}}_t$ (Eq. 5) satisfies condition in Eq. 11,

$$\mathbf{0} \preceq \mathbf{K}_t - \widetilde{\mathbf{K}}_t \preceq \frac{\gamma}{1-\varepsilon}\mathbf{K}_t(\mathbf{K}_t + \gamma\mathbf{I})^{-1} \preceq \frac{\gamma}{1-\varepsilon}\mathbf{I}.$$

and the number of columns selected Q_t is such that

$$Q_t \leq 8\overline{q}.$$

Discussion Unlike in the batch setting, where the sampling procedure always returned m samples, the number of columns Q_t selected by INK-Oracle is a random variable, but with high probability it will be not much larger than $\overline{q}$. Comparing INK-Oracle to online kernel sparsification methods [19], we see that the number of columns, and therefore the space requirement, is guaranteed to be small not only asymptotically but at each step, and that no assumption on the spectrum of the matrix is required. Instead, the space complexity naturally scales with the effective dimension of the problem, and old samples that become superfluous are automatically discarded. Comparing Thm. 1 to Prop. 1, INK-Oracle achieves the same performance as its batch counterpart, as long as the space budget $\overline{q}$ is large enough. This budget depends on several quantities that are difficult to estimate, such as the effective dimension of the full kernel matrix. In practice, this quantity can be interpreted as the maximum amount of space that the user can afford for the algorithm to run. If the actual complexity of the problem exceeds this budget, the user can choose to run it again with another parameter γ or a worse accuracy ε. It is important to notice that, as we show in the proof, the distribution induced by the sampling procedure of INK-Oracle is not the same as the distribution obtained by the multinomial sampling of Batch-Exact. Nonetheless, in our analysis we show that the bias introduced by the different distribution is small, and this allows INK-Oracle to match the approximation guarantees given by Alaoui and Mahoney [1].

We give a detailed proof of Thm. 1 in App. B. In the rest of this section we sketch the proof and give the intuition for the most relevant parts.

The Shrink step uses the thresholding condition to guarantee that the weight $b_{i,t}$ are good approximations of the $\widetilde{p}_{i,t}$. To make the condition effective, we require that the approximate probabilities $\widetilde{p}_{j,t}$ are decreasing. Because the approximate probabilities follow the true probabilities $p_{i,t}$, we first show that this decrease happens for the exact case.

Lemma 1. *For any kernel matrix $\mathbf{K}_t$ at time t, and its bordering $\mathbf{K}_{t+1}$ at time $t+1$ we have that the probabilities $p_{i,t}$ are monotonically decreasing over time t,*

$$\frac{\tau_{i,t+1}}{d_{\mathit{eff}}(\gamma)_{t+1}} = p_{i,t+1} \leq p_{i,t} = \frac{\tau_{i,t+1}}{d_{\mathit{eff}}(\gamma)_t}.$$

Since ridge leverage scores represent the importance of a column, when a new column arrives, there are two cases that can happen. If the column is orthogonal to the existing matrix, none of the previous leverage scores changes. If the new column can explain part of the previous columns, the previous columns should be picked less often, and we expect $\tau_{i,t}$ to decrease. Contrary to RLS, the effective dimension increases when the new sample is orthogonal to the existing matrix, while it stays the same when the new sample is a linear combination of the existing ones. In addition, the presence of γ regularizes both cases. When the vector is nearly orthogonal, the presence of $\gamma\mathbf{I}$ in the inverse will still penalize it, while the γ term at the denominator of Δ will reduce the influence of linearly correlated samples. Because $\tau_{i,t}$ decreases over time and $d_{\text{eff}}(\gamma)_{i,t}$ increases, the probabilities $p_{i,t}$ will overall decrease over time. This result itself is not sufficient to guarantee a well defined `Shrink` step. Due to the (α,β)-approximation, it is possible that $p_{i,t+1} \leq p_{i,t}$ but $\widetilde{p}_{i,t+1} \not\leq \widetilde{p}_{i,t}$. To exclude this possibility, we adapt the following idea from Kelner and Levin [9].

Proposition 2 (Kelner and Levin [9])**.** *Given the approximate probabilities $\widetilde{\mathbf{p}}_t$ returned by an (α,β)-`oracle` at time t, and the approximate probabilities $\widetilde{\mathbf{p}}_{t+1}$ returned by an (α,β)-`oracle` at time $\{t+1\}$, then the approximate probabilities $\min^3\{\widetilde{\mathbf{p}}_t, \widetilde{\mathbf{p}}_{t+1}\}$ are also (α,β)-approximate for $\{t+1\}$. Therefore, without loss of generality, we can assume that $\widetilde{p}_{i,t+1} \leq \widetilde{p}_{i,t}$.*

Combining Lem. 1 and Prop. 2, we can guarantee that at each step the $\widetilde{p}_{i,t}$-s decrease. Unlike in the batch setting [1], we have to take additional care to consider correlations between iterations, the fact that the inclusion probabilities of Alg. 2 are different from the multinomial ones of `Direct-sample`, and that the number of columns kept at each iteration is a *random* quantity Q_t. We adapt the approach of Pachocki [13] to the KRR setting to analyse this process. The key aspect is that the reweighting and rejection rule on line 3 of Alg. 2 will only happen when the probabilities are truly changing. Finally, using a concentration inequality, we show that the number Q_t of columns selected is with high probability only a constant factor away from the budget $\overline{q}$ given to the algorithm.

4 LEVERAGE SCORES AND EFFECTIVE DIMENSION ESTIMATION

In the previous section we showed that our incremental sampling strategy based on (estimated) RLSs has strong space and approximation guarantees for $\widetilde{\mathbf{K}}_n$. While the analysis reported in the previous section relied on the existence of an (α,β)-`oracle` returning accurate leverage scores and effective dimension estimates, in this section we show that such an oracle *exists and can be implemented efficiently.* This is obtained by *two separate estimators* for the RLSs and effective dimension that are updated incrementally and combined together to determine the sampling probabilities.

4.1 Leverage Scores

We start by constructing an estimator that at each time t, takes as input an approximate kernel matrix $\widetilde{\mathbf{K}}_t$, and returns α-approximate RLS $\widetilde{\tau}_{i,t+1}$. The incremental nature of the estimator lies in the fact that it exploits access to the columns already in $\mathbf{S}_t$ and the new (exact) column $\overline{\mathbf{k}}_{t+1}$. We give the following approximation guarantees.

Lemma 2. *We assume $\widetilde{\mathbf{K}}_t$ satisfies Eq. 11 and define $\overline{\mathbf{K}}_{t+1}$ as the matrix bordered with the new row and column*

$$\overline{\mathbf{K}}_{t+1} = \left[\begin{array}{c|c} \widetilde{\mathbf{K}}_t & \overline{\mathbf{k}}_{t+1} \\ \hline \overline{\mathbf{k}}_{t+1}^\intercal & k_{t+1} \end{array}\right].$$

Then

$$\mathbf{0} \preceq \mathbf{K}_{t+1} - \overline{\mathbf{K}}_{t+1} \preceq \frac{\gamma}{1-\varepsilon}\mathbf{I}.$$

Moreover let $\alpha = \frac{2-\varepsilon}{1-\varepsilon}$ and

$$\widetilde{\tau}_{i,t+1} = \frac{1}{\alpha\gamma}\left(k_{i,i} - \mathbf{k}_{i,t+1}\left(\overline{\mathbf{K}}_{t+1} + \alpha\gamma\mathbf{I}\right)^{-1}\mathbf{k}_{i,t+1}\right). \tag{12}$$

Then, for all i such that $\mathbf{k}_{i,t+1} \in \mathcal{I}_t \cup \{t+1\}$,

$$\frac{1}{\alpha}\tau_{i,t+1}(\gamma) \leq \widetilde{\tau}_{i,t+1} \leq \tau_{i,t+1}(\gamma).$$

Remark There are two important details that are used in proof of Lem. 2 (App. C). First, notice that using $\widetilde{\mathbf{K}}_t$ to approximate RLSs directly, would not be accurate enough. RLSs are defined as $\tau_{i,t}(\gamma) = \mathbf{e}_i^\intercal\mathbf{K}_t(\mathbf{K}_t+\gamma\mathbf{I})^{-1}\mathbf{e}_i$ and while the product $(\mathbf{K}_t+\gamma\mathbf{I})^{-1}\mathbf{e}_i$ can be accurately reconstructed using $(\widetilde{\mathbf{K}}_t+\gamma\mathbf{I})^{-1}\mathbf{e}_i$, the multiplication $\mathbf{K}_t\mathbf{e}_i$ cannot be approximated well using $\widetilde{\mathbf{K}}_t$. Since the nullspace of $\widetilde{\mathbf{K}}_t$ can be larger than the one of $\mathbf{K}_t$, it is possible that $\mathbf{e}_i$ partially falls into it, thus compromising the accuracy of the approximation of the RLS. In our approach, we deal with this problem by using the *actual columns* $\mathbf{k}_{i,t}$ of $\mathbf{K}_t$ to compute the RLS. This way, we preserve as much as exact information of the matrix as possible, while the expensive inversion operation is performed on the smaller approximation $\widetilde{\mathbf{K}}_t$. Since we require access to the stored columns $\mathbf{k}_{i,t}$, our approach can approximate the RLSs only for columns present in the dictionary but this is enough, since we are only interested in accurate probabilities for columns in the dictionary and for the new column $\overline{\mathbf{k}}_{t+1}$ (which is available at time $t+1$). As a comparison, the

[3] element-wise mininum

two-pass approach of Alaoui and Mahoney [1] uses the first pass just to compute an approximation $\widetilde{\mathbf{K}}_n$, and then approximates all leverage scores with $\widetilde{\mathbf{K}}_n(\widetilde{\mathbf{K}}_n + \gamma\mathbf{I})^{-1}$. This has an impact on their approximation factor α, that is proportional to $(\lambda_{\min}(\mathbf{K}_n) - \gamma\varepsilon)$. Therefore to have $\alpha \approx (\lambda_{\min}(\mathbf{K}_n) - \gamma\varepsilon) > 0$, it is necessary that $\gamma\varepsilon$ is of the order of $\lambda_{\min}(\mathbf{K}_n)$, which in some cases can be very small, and strongly increase the space requirements of the algorithm. Using the actual columns of the matrix in Eq. 12 allows us to compute an α-approximation independent of the smallest eigenvalue.

4.2 Effective Dimension

Using Eq. 12, we can estimate all the RLSs that we need to update $\mathbf{S}_t$. Nonetheless, to prove that the number of columns selected is not too large, the proof of Thm. 1 in the appendix requires that the sum of the probabilities $\widetilde{p}_{i,t}$ is smaller than 1. Therefore we not only need to compute the RLSs, but also a normalization constant. Indeed, a naïve definition of the probability $\widetilde{p}_{i,t}$ could be $p_{i,t} = \frac{\widetilde{\tau}_{i,t}}{\sum_{j=1}^{t}\widetilde{\tau}_{i,t}}$. A major challenge in our setting is that we cannot compute the sum of the approximate RLSs, because we do not have access to all the columns. Fortunately, we know that $\sum_{j=1}^{t}\widetilde{\tau}_{i,t} \leq \sum_{j=1}^{t}\tau_{i,t}(\gamma) = d_{\text{eff}}(\gamma)_t$. Therefore, one of our technical contribution is an estimator $\widetilde{d}_{\text{eff}}(\gamma)_t$ that does not use the approximate RLSs for the the columns that we no longer have. We now define this estimator and state its approximation accuracy.

Lemma 3. *Assume $\widetilde{\mathbf{K}}_t$ satisfies Eq. 11. Let $\alpha = \left(\frac{2-\varepsilon}{1-\varepsilon}\right)$ and $\beta = \left(\frac{2-\varepsilon}{1-\varepsilon}\right)^2(1+\rho)$ with $\rho = \frac{\lambda_{\max}(\mathbf{K}_n)}{\gamma}$. Define*

$$\widetilde{d}_{\mathit{eff}}(\gamma)_{t+1} = \widetilde{d}_{\mathit{eff}}(\gamma)_t + \alpha\widetilde{\Delta}_t \tag{13}$$

with

$$\begin{aligned}\widetilde{\Delta}_t =& \frac{1}{k_{t+1} + \gamma - \overline{\mathbf{k}}_{t+1}^{\mathsf{T}}\left(\widetilde{\mathbf{K}}_t + \alpha\gamma\mathbf{I}\right)^{-1}\overline{\mathbf{k}}_{t+1}} \\ &\times \left(k_{t+1} - \overline{\mathbf{k}}_{t+1}^{\mathsf{T}}\left(\widetilde{\mathbf{K}}_t + \alpha\gamma\mathbf{I}\right)^{-1}\overline{\mathbf{k}}_{t+1}\right. \\ &\left. - \frac{(1-\varepsilon)^2}{4}\gamma\overline{\mathbf{k}}_{t+1}^{\mathsf{T}}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})^{-2}\overline{\mathbf{k}}_{t+1}\right). \end{aligned} \tag{14}$$

Then

$$d_{\mathit{eff}}(\gamma)_{t+1} \leq \widetilde{d}_{\mathit{eff}}(\gamma)_{t+1} \leq \beta d_{\mathit{eff}}(\gamma)_{t+1}.$$

Discussion Since we cannot compute accurate RLSs for columns that are not present in the dictionary, we prefer to not estimate how each RLSs changes over time, but instead we directly estimate the increment of their sum. We do it by updating our estimate $\widetilde{d}_{\text{eff}}(\gamma)_{t+1}$ using our previous estimate $\widetilde{d}_{\text{eff}}(\gamma)_t$, and $\widetilde{\Delta}_t$. $\widetilde{\Delta}_t$ captures directly the interaction of the new sample with the *aggregate* of the previous samples, and allows us to estimate the increase in effective dimension using only the current matrix approximation $\widetilde{\mathbf{K}}_t$, the new column $\overline{\mathbf{k}}_{t+1}$ and the scalar k_{t+1}. Differently from the other terms we studied, the numerator of $\widetilde{\Delta}_t$ contains an additional $\gamma\overline{\mathbf{k}}_{t+1}^{\mathsf{T}}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})^{-2}\overline{\mathbf{k}}_{t+1}$ second order term. The guarantees provided by Eq. 11 are not straightforward to extend because in general if $(\mathbf{K}_t + \gamma\mathbf{I})^{-1} \succeq (\widetilde{\mathbf{K}}_t + \alpha\gamma\mathbf{I})^{-1}$, it is not guaranteed that $(\mathbf{K}_t + \gamma\mathbf{I})^{-2} \succeq (\widetilde{\mathbf{K}}_t + \alpha\gamma\mathbf{I})^{-2}$. Nonetheless, we show that $\widetilde{\mathbf{K}}_t$ is still sufficient to estimate $\widetilde{\Delta}_t$, but, unlike α, the approximation error β is now dependent on the spectrum.

Algorithm 3 The `INK-Estimate` algorithm

Input: Dataset $\mathcal{D}$, regularization γ, sampling budget $\overline{q}$

Output: $\widetilde{\mathbf{K}}_n$, $\mathbf{S}_n$

Initialize $\mathcal{I}_0$ as empty, $\widetilde{p}_{1,0} = 1, b_{1,0} = 1$, budget $\overline{q}$
for $t = 0, \dots, n-1$ **do**
  Receive new column $\overline{\mathbf{k}}_{t+1}$ and scalar k_{t+1}
  Compute α-leverage scores $\{\widetilde{\tau}_{i,t+1} : i \in \mathcal{I}_t \cup \{t+1\}\}$, using $\overline{\mathbf{K}}_{t+1}$, $\mathbf{k}_i$, $k_{i,i}$, and Eq. 12
  Compute β-approximate $\widetilde{d}_{\text{eff}}(\gamma)_{t+1}$ using $\widetilde{\mathbf{K}}_t$, $\overline{\mathbf{k}}_{t+1}$, k_{t+1}, and Eq. 13
  Set $\widetilde{p}_{i,t+1} = \min\{\widetilde{\tau}_{i,t+1}/\widetilde{d}_{\text{eff}}(\gamma)_{t+1}, \widetilde{p}_{i,t}\}$
  $\mathcal{I}_{t+1}, \mathbf{b}_{t+1}$ = `Shrink-Expand`$(\mathcal{I}_t, \widetilde{\mathbf{p}}_{t+1}, \mathbf{b}_t, \overline{q})$
  Compute $\mathbf{S}_{t+1}$ using $\mathcal{I}_{t+1}$ and weights $\sqrt{b_{i,t+1}}$
  Compute $\widetilde{\mathbf{K}}_{t+1}$ using $\mathbf{S}_{t+1}$ and Eq. 5
end for
Return $\widetilde{\mathbf{K}}_n$ and $\mathbf{S}_n$

4.3 Analysis of `INK-Estimate`

With the separate estimates for leverage scores (Sect. 4.1) and effective dimension (Sect. 4.2), we have the necessary ingredients for the (α, β)-`oracle` and we are ready to present the final algorithm `INK-Estimate` (Alg. 3).

Using the approximation guarantees of Lem. 2 and Lem. 3, we are ready to state the final result, instantiating the generic α and β terms of Thm. 2 with the values obtained in this section.

Theorem 2. *Let $\rho = \lambda_{\max}(\mathbf{K}_t)/\gamma$, $\alpha = \left(\frac{2-\varepsilon}{1-\varepsilon}\right)$, $\beta = \left(\frac{2-\varepsilon}{1-\varepsilon}\right)^2(1+\rho)$, and $\gamma > 1$. For any $0 \leq \varepsilon \leq 1$, and $0 \leq \delta \leq 1$, if we run Alg. 3 with parameter $\overline{q}$, where*

$$\overline{q} \geq \left(\frac{28\alpha\beta d_{\mathit{eff}}(\gamma)_t}{\varepsilon^2}\right)\log\left(\frac{4t}{\delta}\right),$$

to compute a sequence of random matrices $\mathbf{S}_t$ with a random number of columns Q_t, then with probability $1-\delta$, for all t the corresponding Nyström approximation $\widetilde{\mathbf{K}}_t$ (Eq. 5)

satisfies (11)

$$\mathbf{0} \preceq \mathbf{K}_t - \widetilde{\mathbf{K}}_t \preceq \frac{\gamma}{1-\varepsilon}\mathbf{K}_t(\mathbf{K}_t+\gamma\mathbf{I})^{-1} \preceq \frac{\gamma}{1-\varepsilon}\mathbf{I}.$$

With the same prob., `INK-Estimate` *requires at most*

$$\begin{aligned}\mathcal{O}(n^2\bar{q}^2 + n\bar{q}^3) &\leq \mathcal{O}\left(\alpha^2\beta^2 n^2 d_{\mathit{eff}}(\gamma)_n^2 + \alpha^3\beta^3 n d_{\mathit{eff}}(\gamma)_n^3\right)\\ &= \mathcal{O}\left(\alpha^4(1+\rho)^2 n^2 d_{\mathit{eff}}(\gamma)_n^2 + \alpha^6(1+\rho)^3 n d_{\mathit{eff}}(\gamma)_n^3\right)\end{aligned}$$

time and the space is bounded as

$$\mathcal{O}(n\bar{q}) \leq \mathcal{O}\left(\alpha\beta n d_{\mathit{eff}}(\gamma)_n\right) = \mathcal{O}\left(\alpha^2(1+\rho) n d_{\mathit{eff}}(\gamma)_n\right).$$

For the space complexity, from Thm. 1 we know we will not select more than $\mathcal{O}(\bar{q})$ columns in high probability. For the time complexity, at each iteration we need to solve linear systems involving $(\overline{\mathbf{K}}_{t+1}+\alpha\gamma\mathbf{I})^{-1}$ and $(\widetilde{\mathbf{K}}_t+\alpha\gamma\mathbf{I})^{-1}$. Approximating the inverse using transformations similar to Eq. 6 takes $\mathcal{O}(t\bar{q}^2+\bar{q}^3)$ time, again using a singular value decomposition approach. To compute all leverage scores, we need to first compute an approximate inverse in $\mathcal{O}(t\bar{q}^2+\bar{q}^3)$ time, and then solve Q_t systems, each using a multiplication costing $\mathcal{O}(tQ_t)$. With high probability, $Q_t \leq 8\bar{q}$, therefore computing all leverage scores costs $\mathcal{O}(t\bar{q}^2+\bar{q}^3)$ for the first singular value decomposition, and $\mathcal{O}(t\bar{q})$ for each of the $\mathcal{O}(\bar{q})$ applications. To update the effective dimension estimate, we only have to compute another approximate inverse, and that costs $\mathcal{O}(t\bar{q}^2+\bar{q}^3)$ as well. Finally, we have to sum the costs over n steps, and from $\sum_{t=1}^n t\bar{q}^2 \leq \bar{q}^2 n^2$, we obtain the final complexity. Even with a significantly different approach, `INK-Estimate` achieves the same approximation guarantees as `Batch-Exact`. Consequently, it provides the same risk guarantees as the known batch version [1], stated in the following corollary.

Corollary 1. *For every $t \in \{1,\dots,n\}$, let $\mathbf{K}_t$ be the kernel matrix at time t. Run Alg. 3 with regularization parameter γ and space budget $\bar{q}$. Then, at any time t, the solution $\widetilde{\mathbf{w}}_t$ computed using the regularized Nyström approximation $\widetilde{\mathbf{K}}_t$ satisfies*

$$\begin{aligned}\mathcal{R}(\widetilde{\mathbf{w}}_t) &\leq \left(1+\frac{\gamma}{\mu}\frac{1}{1-\varepsilon}\right)^2 \mathcal{R}(\widehat{\mathbf{w}}_t)\\ &= \left(1+\frac{\lambda_{\max}(\mathbf{K}_t)}{\rho\mu}\frac{1}{1-\varepsilon}\right)^2 \mathcal{R}(\widehat{\mathbf{w}}_t).\end{aligned}$$

Discussion Thm. 2 combines the generic result of Thm. 1 with the actual implementation of an oracle that we developed in this section. All the guarantees that hold for `INK-Oracle` are inherited by `INK-Estimate`, but now we can quantify the impact of the errors α and β on the algorithm. As we saw, the α error does not depends on the time, the spectrum of the kernel matrix or other quantities that increase over time. On the other hand, estimating the effective dimension without having access to all the leverage scores is a much harder task, and the β factor depends on the spectrum through the ρ coefficient. The influence that this coefficient exerts on the space and time complexity can vary significantly as the relative magnitude of $\lambda_{\max}(\mathbf{K}_n)$, γ and μ changes. If the largest eigenvalue grows too large without a corresponding increase in γ, the space and time requirements of `INK-Estimate` can grow, but the risk bound, depending on γ/μ remains small. On the other hand, increasing γ without increasing μ reduces the computational complexity, but makes the guarantees on the risk of the solution $\widetilde{\mathbf{w}}_t$ much weaker. As an example, Alaoui and Mahoney [1, Thm. 3] choose, $\mu \geq \lambda_{\max}(\mathbf{K}_n)$ and $\gamma \approxeq \mu$. If we do the same, we recover their bound.

5 CONCLUSION

We presented a space-efficient algorithm for sequential Nyström approximation that requires only a single pass over the dataset to construct a low-rank matrix $\widetilde{\mathbf{K}}_n$ that accurately approximates the kernel matrix $\mathbf{K}_n$, and compute an approximate KRR solution $\widetilde{\mathbf{w}}_n$ whose risk is close to the exact solution $\widehat{\mathbf{w}}_n$. All of these guarantees do not hold only for the final matrix, but are valid for all intermediate matrices $\widetilde{\mathbf{K}}_t$ constructed by the sequential algorithm.

To address the challenges coming from the sequential setup, we introduced two separate estimators for RLSs and effective dimension that provide multiplicative error approximations of these two quantities across iterations. While the approximation of the RLSs is only a constant factor away from the exact RLSs, the error of the approximate effective dimension scales with the spectrum of the matrix through the coefficient ρ. A more careful analysis, or a different estimator might improve this dependence, and they can be easily plugged to the general analysis.

Our generalization results apply to the fixed design setting. An important extension of our work would be to consider a random design, such as in the work of Rudi et al. [16]. This extension would need even more careful tuning of the regularization parameter γ, needing to satisfy requirements of both generalization and the approximation of the (α,β)-`oracle`. Finally, the runtime analysis of the algorithm does not fully exploit the sequential nature of the updates. An implementation based on decompositions more amenable to updates (e.g., Cholesky decomposition), or on low-rank solvers that can exploit hot-start might further improve the time complexity.

Acknowledgements The research presented in this paper was supported by CPER Nord-Pas de Calais/FEDER DATA Advanced data science and technologies 2015-2020, French Ministry of Higher Education and Research, Nord-Pas-de-Calais Regional Council, French National Research Agency project ExTra-Learn (n.ANR-14-CE24-0010-01).

References

[1] Ahmed El Alaoui and Michael W. Mahoney. Fast randomized kernel methods with statistical guarantees. In *Neural Information Processing Systems*, 2015.

[2] Francis Bach. Sharp analysis of low-rank kernel matrix approximations. In *International Conference on Learning Theory*, 2013.

[3] Petros Drineas, Malik Magdon-Ismail, Michael W Mahoney, and David P. Woodruff. Fast approximation of matrix coherence and statistical leverage. *International Conference on Machine Learning*, 2012.

[4] Yaakov Engel, Shie Mannor, and Ron Meir. Sparse online greedy support vector regression. In *European Conference on Machine Learning*, 2002.

[5] Yaakov Engel, Shie Mannor, and Ron Meir. The kernel recursive least-squares algorithm. *IEEE Transactions on Signal Processing*, 52(8):2275–2285, 2004.

[6] B. S. Everitt. *The Cambridge dictionary of statistics*. Cambridge University Press, Cambridge, 2002.

[7] Alex Gittens and Michael Mahoney. Revisiting the Nyström method for improved large-scale machine learning. In *International Conference on Machine Learning*.

[8] Nicholas J Higham. *Accuracy and stability of numerical algorithms*. Society for Industrial and Applied Mathematics, 2002.

[9] Jonathan A. Kelner and Alex Levin. Spectral sparsification in the semi-streaming setting. *Theory of Computing Systems*, 53(2):243–262, 2012.

[10] Jyrki Kivinen, Alexander J. Smola, and Robert C. Williamson. Online learning with kernels. *IEEE Transactions on Signal Processing*, 52(8):2165–2176, 2004.

[11] Quoc Le, Tamás Sarlós, and Alex J Smola. Fastfood — Approximating kernel expansions in loglinear time. In *International Conference on Machine Learning*, 2013.

[12] Weifeng Liu, Il Park, and Jose C. Principe. An information theoretic approach of designing sparse kernel adaptive filters. *IEEE Transactions on Neural Networks*, 20(12):1950–1961.

[13] Jakub Pachocki. Analysis of resparsification. *arXiv preprint arXiv:1605.08194*, 2016.

[14] Ali Rahimi and Ben Recht. Random features for large-scale kernel machines. In *Neural Information Processing Systems*, 2007.

[15] Cédric Richard, José Carlos M. Bermudez, and Paul Honeine. Online prediction of time series data with kernels. *IEEE Transactions on Signal Processing*, 57 (3):1058–1067.

[16] Alessandro Rudi, Raffaello Camoriano, and Lorenzo Rosasco. Less is more: Nyström computational regularization. In *Neural Information Processing Systems*, 2015.

[17] Bernhard Schölkopf and Alexander J. Smola. *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT Press, 2001.

[18] John Shawe-Taylor and Nelo Cristianini. *Kernel methods for pattern analysis*. Cambridge University Press, 2004.

[19] Yi Sun, Jürgen Schmidhuber, and Faustino J. Gomez. On the size of the online kernel sparsification dictionary. In *International Conference on Machine Learning*, 2012.

[20] Joel A Tropp. Freedman's inequality for matrix martingales. *Electron. Commun. Probab*, 16:262–270, 2011.

[21] Joel A. Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends in Machine Learning*, 8(1-2):1–230, 2015.

[22] Steven Van Vaerenbergh, Miguel Lázaro-Gredilla, and Ignacio Santamaría. Kernel recursive least-squares tracker for time-varying regression. *Neural Networks and Learning Systems, IEEE Transactions on*, 23(8):1313–1326, 2012.

[23] Christopher Williams and Matthias Seeger. Using the Nystrom method to speed up kernel machines. In *Neural Information Processing Systems*, 2001.

[24] Yuchen Zhang, John C. Duchi, and Martin J. Wainwright. Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates. *Journal Machine Learning Research*, 16:3299–3340, 2015.

A Extended proofs of Section 3

Proof of Lemma 1. The proof follows from studying the evolution of the numerator and denominator (i.e., RLSs and effective dimension) separately.

Lemma 4. *For any kernel matrix $\mathbf{K}_t$ and its bordering $\mathbf{K}_{t+1}$, we have for all $i = \{1, \dots, t\}$*

$$\tau_{i,t}(\gamma) \geq \tau_{i,t+1}(\gamma)$$

Proof of Lemma 4. We want to show that for all $i = \{1, \dots, t\}$

$$\frac{\tau_{i,t+1}(\gamma)}{\tau_{i,t}(\gamma)} = \frac{\mathbf{e}_{i,t+1}^\mathsf{T} \mathbf{K}_{t+1} (\mathbf{K}_{t+1}^\gamma)^{-1} \mathbf{e}_{i,t+1}}{\mathbf{e}_{i,t}^\mathsf{T} \mathbf{K}_t (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_{i,t}} \leq 1,$$

where $\mathbf{K}_t^\gamma = \mathbf{K}_t + \gamma \mathbf{I}_t$. Since $\mathbf{K}_{t+1}$ is obtained as the bordering of $\mathbf{K}_t$ using the vector $\overline{\mathbf{k}}_{t+1} \in \mathbb{R}^t$ and the element k_{t+1} (see Eq. 1), we can use the block matrix inverse formula and obtain

$$(\mathbf{K}_{t+1}^\gamma)^{-1} = \left[\begin{array}{c|c} (\mathbf{K}_t^\gamma) & \overline{\mathbf{k}}_{t+1} \\ \hline \overline{\mathbf{k}}_{t+1}^\mathsf{T} & k_{t+1} + \gamma \end{array}\right]^{-1} = \frac{1}{\xi} \left[\begin{array}{c|c} \xi (\mathbf{K}_t^\gamma)^{-1} + (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} \overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} & -(\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} \\ \hline -\overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} & 1 \end{array}\right],$$

with

$$\xi = k_{t+1} + \gamma - \overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1}. \tag{15}$$

Then we can rewrite the ratio between the RLSs as

$$\begin{aligned} \frac{\tau_{i,t+1}(\gamma)}{\tau_{i,t}(\gamma)} &= \frac{\mathbf{e}_{i,t+1}^\mathsf{T} \mathbf{K}_{t+1} (\mathbf{K}_{t+1}^\gamma)^{-1} \mathbf{e}_{i,t+1}}{\mathbf{e}_{i,t}^\mathsf{T} \mathbf{K}_t (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_{i,t}} \\ &= \frac{[\mathbf{k}_{i,t}^\mathsf{T}, \ \mathcal{K}(x_i, x_{t+1})] \ \cdot \ [\xi (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_{i,t} + (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} \overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_{i,t}, \ -\overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_{i,t}]^\mathsf{T}}{\xi \mathbf{k}_{i,t}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_{i,t}}. \end{aligned}$$

where $\mathbf{k}_{i,t} = \mathbf{K}_t \mathbf{e}_{i,t}$. In the following we drop the dependency on t from indicator vectors and kernel vectors and we write $\mathbf{k}_i = \mathbf{k}_{i,t}$ and $\mathbf{e}_i = \mathbf{e}_{i,t}$. As a result we can further rewrite the previous expression as

$$\begin{aligned} \frac{\tau_{i,t+1}(\gamma)}{\tau_{i,t}(\gamma)} &= \frac{\xi \mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i + \mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} \overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i - \mathcal{K}(x_i, x_{t+1}) \overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i}{\xi \mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i} \\ &= 1 - \frac{\overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i (\mathcal{K}(x_i, x_{t+1}) - \mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1})}{\xi \mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i}. \end{aligned}$$

We first focus on the term in parenthesis at the numerator. Since by definition $\mathcal{K}(x_i, x_{t+1}) = \mathbf{e}_i^\mathsf{T} \overline{\mathbf{k}}_{t+1}$, we have

$$\begin{aligned} &\mathcal{K}(x_i, x_{t+1}) - \mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} = \mathbf{e}_i^\mathsf{T} \overline{\mathbf{k}}_{t+1} - \mathbf{e}_i^\mathsf{T} \mathbf{K}_t (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} \\ &= \mathbf{e}_i^\mathsf{T} \overline{\mathbf{k}}_{t+1} - \mathbf{e}_i^\mathsf{T} (\mathbf{K}_t + \gamma \mathbf{I}) (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} + \gamma \mathbf{e}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} \\ &= \mathbf{e}_i^\mathsf{T} \overline{\mathbf{k}}_{t+1} - \mathbf{e}_i^\mathsf{T} (\mathbf{K}_t^\gamma)(\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} + \gamma \mathbf{e}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1} = \gamma \mathbf{e}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \overline{\mathbf{k}}_{t+1}. \end{aligned}$$

Thus the ratio can be finally written as

$$\frac{\tau_{i,t+1}(\gamma)}{\tau_{i,t}(\gamma)} = 1 - \frac{\gamma (\overline{\mathbf{k}}_{t+1}^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i)^2}{\xi \mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i}.$$

Since $\gamma > 0$, we only need to analyze the sign of the denominator to prove the final statement. Since $\mathbf{k}_i^\mathsf{T} (\mathbf{K}_t^\gamma)^{-1} \mathbf{e}_i = \tau_{i,t} > 0$ by definition of RLS, the only term which needs some care is the coefficient ξ of the block matrix inverse formula. As illustrated in Sect. 2.1, the kernel function applied to any to points is $\mathcal{K}(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle$, where φ is the feature map. As a result, the kernel matrix and kernel vector can be written as $\mathbf{K}_t = \mathbf{\Phi}_t^\mathsf{T} \mathbf{\Phi}_t$ and $\overline{\mathbf{k}}_{t+1} = \mathbf{\Phi}_t^\mathsf{T} \varphi(x_{t+1})$, where

$\mathbf{\Phi}_t \in \mathbb{R}^{D\times t}$. Let $\mathbf{\Phi}_t = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^\top$ be the SVD decomposition of $\mathbf{\Phi}$, with $\mathbf{V} \in \mathbb{R}^{D\times D}$ and $\mathbf{U} \in \mathbb{R}^{t\times t}$ are orthonormal and $\mathbf{\Sigma} \in \mathbb{R}^{D\times t}$ is a rectangular diagonal matrix with singular values σ_i on the diagonal. We have

$$\begin{aligned}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t + \gamma\mathbf{I})\overline{\mathbf{k}}_{t+1} &= \varphi(x_{t+1})^\top\mathbf{\Phi}_t(\mathbf{\Phi}_t^\top\mathbf{\Phi}_t + \gamma\mathbf{I})^{-1}\mathbf{\Phi}_t^\top\varphi(x_{t+1}) \\ &= \varphi(x_{t+1})^\top\mathbf{V}\mathbf{\Sigma}(\mathbf{\Sigma}^\top\mathbf{\Sigma} + \gamma\mathbf{I})^{-1}\mathbf{\Sigma}^\top\mathbf{V}^\top\varphi(x_{t+1}).\end{aligned}$$

The central term $\mathbf{\Sigma}(\mathbf{\Sigma}^\top\mathbf{\Sigma} + \gamma\mathbf{I})^{-1}\mathbf{\Sigma}^\top$ is still a rectangular diagonal matrix with elements $\frac{\sigma_i^2}{\sigma_i^2+\gamma}$ for $i \in [t]$. With an abuse of notation we denote it as $\mathrm{Diag}\left(\left\{\frac{\sigma_i^2}{\sigma_i^2+\gamma}\right\}_i\right)$. Then we can write the previous expression as

$$\begin{aligned}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t + \gamma\mathbf{I})\overline{\mathbf{k}}_{t+1} &= \varphi(x_{t+1})^\top\mathbf{V}\,\mathrm{Diag}\left(\left\{\frac{\sigma_i^2}{\sigma_i^2+\gamma}\right\}_i\right)\mathbf{V}^\top\varphi(x_{t+1}) \\ &\leq \varphi(x_{t+1})^\top\mathbf{V}\mathbf{I}\mathbf{V}^\top\varphi(x_{t+1}) = \varphi(x_{t+1})^\top\varphi(x_{t+1}) = \mathcal{K}(x_{t+1}, x_{t+1}) = k_{t+1},\end{aligned}$$

which implies that

$$\xi = k_{t+1} + \gamma - \overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1} \geq \gamma > 0,$$

thus concluding the proof.

□

Lemma 5. *For any kernel matrix $\mathbf{K}_t$ and its bordering $\mathbf{K}_{t+1}$, we have*

$$d_{\mathit{eff}}(\gamma)_{t+1} = d_{\mathit{eff}}(\gamma)_t + \Delta \geq d_{\mathit{eff}}(\gamma)_t,$$

where

$$\Delta = \frac{k_{t+1} - \overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1} - \gamma\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-2}\overline{\mathbf{k}}_{t+1}}{\xi} \geq 0 \tag{16}$$

and

$$\xi = k_{t+1} + \gamma - \overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}. \tag{17}$$

Proof of Lemma 5. We first derive the incremental update of the effective dimension as

$$\begin{aligned}&d_{\text{eff}}(\gamma)_{t+1} = \mathrm{Tr}\left(\mathbf{K}_{t+1}(\mathbf{K}_{t+1} + \gamma\mathbf{I})^{-1}\right) \\ &= \frac{1}{\xi}\mathrm{Tr}\left(\left[\begin{array}{c|c}\mathbf{K}_t & \overline{\mathbf{k}}_{t+1} \\ \hline \overline{\mathbf{k}}_{t+1}^\top & k_{t+1}\end{array}\right]\left[\begin{array}{c|c}\xi(\mathbf{K}_t^\gamma)^{-1} + (\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1} & -(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1} \\ \hline -\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1} & 1\end{array}\right]\right) \\ &= \mathrm{Tr}(\mathbf{K}_t(\mathbf{K}_t^\gamma)^{-1}) + \frac{\mathrm{Tr}(\mathbf{K}_t(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1})}{\xi} - 2\frac{\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}}{\xi} + \frac{k_{t+1}}{\xi} \\ &= d_{\text{eff}}(\gamma)_t + \frac{\mathrm{Tr}((\mathbf{K}_t + \gamma\mathbf{I})(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1})}{\xi} \\ &\qquad - \gamma\frac{\mathrm{Tr}((\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1})}{\xi} - 2\frac{\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}}{\xi} + \frac{k_{t+1}}{\xi} \\ &= d_{\text{eff}}(\gamma)_t + \frac{\mathrm{Tr}(\mathbf{I}\overline{\mathbf{k}}_{t+1}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1})}{\xi} - \gamma\frac{\mathrm{Tr}((\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1})}{\xi} - 2\frac{\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1}}{\xi} + \frac{k_{t+1}}{\xi} \\ &= d_{\text{eff}}(\gamma)_t + \frac{k_{t+1} - \overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1} - \gamma\overline{\mathbf{k}}_{t+1}^\top(\mathbf{K}_t^\gamma)^{-2}\overline{\mathbf{k}}_{t+1}}{\xi},\end{aligned}$$

where we use the bordering of the kernel and the block matrix inversion formula. Using the same arguments as in Lemma 4, we have $\xi > 0$, thus we only need to focus on the numerator of the second term in the previous expression. We use the

feature map matrix $\mathbf{\Phi}_t$ and the fact that $\mathbf{K}_t = \mathbf{\Phi}_t\mathbf{\Phi}_t^\mathsf{T}$ to obtain

$$\begin{aligned}
&k_{t+1} - \overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1} - \gamma\overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t^\gamma)^{-2}\overline{\mathbf{k}}_{t+1} \\
&= \varphi(x_{t+1})^\mathsf{T}\varphi(x_{t+1}) - \varphi(x_{t+1})^\mathsf{T}\mathbf{\Phi}_t(\mathbf{\Phi}_t\mathbf{\Phi}_t^\mathsf{T} + \gamma)^{-1}\mathbf{\Phi}_t^\mathsf{T}\varphi(x_{t+1}) - \gamma\varphi(x_{t+1})^\mathsf{T}\mathbf{\Phi}_t(\mathbf{\Phi}_t\mathbf{\Phi}_t^\mathsf{T} + \gamma)^{-2}\mathbf{\Phi}_t^\mathsf{T}\varphi(x_{t+1}) \\
&= \varphi(x_{t+1})^\mathsf{T}\Big(\mathbf{I} - \mathbf{\Phi}_t(\mathbf{\Phi}_t\mathbf{\Phi}_t^\mathsf{T} + \gamma)^{-1}\mathbf{\Phi}_t^\mathsf{T} - \gamma\mathbf{\Phi}_t(\mathbf{\Phi}_t\mathbf{\Phi}_t^\mathsf{T} + \gamma)^{-2}\mathbf{\Phi}_t^\mathsf{T}\Big)\varphi(x_{t+1}).
\end{aligned}$$

Using the singular value decomposition of the feature matrix as $\mathbf{\Phi}_t = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}$, we can rewrite the central part of the quadratic form above as

$$\begin{aligned}
&k_{t+1} - \overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t^\gamma)^{-1}\overline{\mathbf{k}}_{t+1} - \gamma\overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t^\gamma)^{-2}\overline{\mathbf{k}}_{t+1} \\
&= \phi(x_{t+1})^\mathsf{T}\mathbf{V}\,\mathrm{Diag}\left(\left\{1 - \frac{\sigma_i^2}{\sigma_i^2+\gamma} - \frac{\gamma\sigma_i^2}{(\sigma_i^2+\gamma)^2}\right\}_{i=1}^t\right)\mathbf{V}^\mathsf{T}\phi(x_{t+1}) \\
&= \varphi(x_{t+1})^\mathsf{T}\mathbf{V}\,\mathrm{Diag}\left(\left\{\left(\frac{\gamma}{\sigma_i^2+\gamma}\right)^2\right\}_{i=1}^d\right)\mathbf{V}^\mathsf{T}\varphi(x_{t+1}) > 0,
\end{aligned}$$

which guarantees that the increment to the effective dimension is non-negative and concludes the proof. □

Combining the two lemmas, we obtain Lemma 1. □

Proof of Proposition 2. With a slight abuse of notation, for this proof we indicate with $\mathbf{p}_{t+1} \leq \mathbf{p}_t$ that for each $i \in \{1, \dots, t\}$ we have $p_{i,t+1} \leq p_{i,t}$. We know that $\widetilde{\mathbf{p}}_t \leq \mathbf{p}_t$ and $\widetilde{\mathbf{p}}_{t+1} \leq \mathbf{p}_{t+1}$. Then obviously

$$\min\{\widetilde{\mathbf{p}}_t, \widetilde{\mathbf{p}}_{t+1}\} \leq \min\{\mathbf{p}_t, \mathbf{p}_{t+1}\} \leq \mathbf{p}_{t+1}$$

For the lower bound, we know from Lemma 1 that $\mathbf{p}_{t+1} \leq \mathbf{p}_t$. Therefore

$$\min\{\widetilde{\mathbf{p}}_t, \widetilde{\mathbf{p}}_{t+1}\} \geq \frac{1}{\alpha\beta}\min\{\mathbf{p}_t, \mathbf{p}_{t+1}\} \geq \frac{1}{\alpha\beta}\mathbf{p}_{t+1}.$$

□

B Proof of Theorem 1 and 2

Proof of Theorem 1 and 2. We will first prove the result for generic (α, β) approximate probabilities, thus proving Theorem 1. Theorem 2 directly follows by placing the correct values for α and β computed in Section 4. We will first prove the result for generic (α, β) approximate probabilities, thus proving Theorem 1. Theorem 2 directly follows by placing the correct values for α and β computed in Section 4. For the proof, we will use the following matrix concentration inequalities.

Proposition 3 (Thm. 1.2 [20])**.** *Consider a matrix martingale $\{\mathbf{Y}_k : k = 0, 1, 2, \dots\}$ whose values are self-adjoint matrices with dimension d, and let $\{\mathbf{X}_k : k = 1, 2, 3, \dots\}$ be the difference sequence. Assume that the difference sequence is uniformly bounded in the sense that*

$$\lambda_{\max}(\mathbf{X}_k) \leq R \quad \textit{almost surely} \quad \textit{for } k = 1, 2, 3, \dots.$$

Define the predictable quadratic variation process of the martingale:

$$\mathbf{W}_k := \sum_{j=1}^k \mathbb{E}\left[\mathbf{X}_j^2 \;\middle|\; \{\mathbf{X}_s\}_{s=0}^{j-1}\right]. \quad \textit{for } k = 1, 2, 3, \dots.$$

Then, for all $t \geq 0$ and $\sigma^2 > 0$,

$$\mathbb{P}\left(\exists k \geq 0 : \lambda_{\max}(\mathbf{Y}_k) \geq t \;\textit{ and }\; \|\mathbf{W}_k\| \leq \sigma^2\right) \leq d \cdot \exp\left\{-\frac{t^2/2}{\sigma^2 + Rt/3}\right\}.$$

Proposition 4 (Thm. 5.1.1 [21]). *Consider a finite sequence $\{\mathbf{V}_j\}$ of independent random symmetric PSD matrices with dimension t. Assume that $\lambda_{\max}(\mathbf{V}_j) \leq R$ and define*

$$\mu = \left\| \sum_j \mathbb{E}\left[\mathbf{V}_j\right] \right\|_2 .$$

then, for all $\delta \geq 0$,

$$\mathbb{P}\left(\lambda_{\max}\left(\sum_j \mathbf{V}_j \right) \geq (1+\delta)\mu \right) \leq t \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \right)^{\mu/R}$$

Let $\mathbf{K}_t = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\mathsf{T}$ be the eigendecomposition of the kernel matrix at time t. Then we define $\mathbf{\Psi}_t = \mathbf{\Lambda}^{1/2}(\mathbf{\Lambda} + \gamma\mathbf{I})^{-1/2}\mathbf{U}^\mathsf{T}$. In the following we drop the dependency on t and we use $\mathbf{\Psi}_t = \mathbf{\Psi}$ and we denote by ψ_i the i-th column of $\mathbf{\Psi}$ so that

$$\mathbf{\Psi}\mathbf{\Psi}^\mathsf{T} = \sum_{i=1}^{t} \psi_i \psi_i^\mathsf{T}.$$

We recall that $\mathbf{\Psi}$, which describes the ratio between the spectrum of the kernel and its soft-thresholded version, is strictly related to both the RLSs and the effective dimension, since $\tau_{i,t}(\gamma) = \|[\mathbf{\Psi}_t]_{:,i}\|_2^2$ and $d_{\text{eff}}(\gamma)_t = \|\mathbf{\Psi}_t^2\|_F$.

We want to show that, for an appropriately chosen $\mathbf{S}_t$, the largest eigenvalue of

$$\mathbf{\Psi}\mathbf{\Psi}^\mathsf{T} - \mathbf{\Psi}\mathbf{S}_t\mathbf{S}_t^\mathsf{T}\mathbf{\Psi}^\mathsf{T} = \sum_{i=1}^{t} \left(1 - b_{i,t}\right) \psi_i \psi_i^\mathsf{T}$$

is small. In particular, Alaoui and Mahoney [1] showed that if $\lambda_{\max}(\mathbf{\Psi}\mathbf{\Psi}^\mathsf{T} - \mathbf{\Psi}\mathbf{S}_t\mathbf{S}_t^\mathsf{T}\mathbf{\Psi}^\mathsf{T}) \leq \varepsilon$, then the approximated matrix $\widetilde{\mathbf{K}}_t$ deterministically satisfies (11).

Approximation accuracy The following proof follows closely the approach introduced in Pachocki [13]. In particular, we will follow the random evolution of the weights $b_{i,s}$ (defined by the behaviour of Subroutine `Shrink-Expand`) as the algorithm runs. The subscript (i, s) indexes these weights temporally, following the iterations of our algorithm, but the relationship between s and the value of $b_{i,s}$ is not immediate. In particular, if $\widetilde{p}_{i,s}$ does not decrease sufficiently at one iteration, $b_{i,s} = b_{i,s-1}$ and the weight of that column stays unchanged. If instead it decreases significantly, the inner loop of `Shrink-Expand` (lines 3 and following) can execute multiple times, and $b_{i,s} - b_{i,s-1}$ can be greater than 1. Nonetheless, if $b_{i,s} = l$, we know that none of the Bernoulli trials $\mathcal{B}\left(\frac{k}{k+1}\right)$ in the sequence $k = \{1, 2, \ldots, l\}$ have failed, because if any of those trials failed we would have set the variable to 0 forever.

We can represent both `INK-Oracle` and `INK-Estimate` as the following random process

$$\widehat{\mathbf{Y}}_{i,s} = \left(\sum_{k=1}^{i} \left(1 - b_{k,s}\right) \psi_k \psi_k^\mathsf{T} \right) + \left(\sum_{k=i+1}^{t} \left(1 - b_{k,s-1}\right) \psi_k \psi_k^\mathsf{T} \right)$$

where the differences $\widehat{\mathbf{X}}_{i,s} = \widehat{\mathbf{Y}}_{i,s} - \widehat{\mathbf{Y}}_{i,s-1}$ are

$$\widehat{\mathbf{X}}_{i,s} = \left(b_{i,s-1} - b_{i,s}\right) \psi_i \psi_i^\mathsf{T},$$

For a consistent notation, we simply set $b_{i,s} = 1$ for all $s < i$. Intuitively, for t time steps, indexed by s, the algorithm loops over t columns, indexed by i. For each columns, it uses a deterministic function $f(\{b_{i,s-1}\}_{i=1}^t)$ to compute $\widetilde{p}_{i,s}$. If $\widetilde{p}_{i,s}$ decreases enough, compared to $\widetilde{p}_{i,s-1}$, it executes one or more independent Bernoulli trials to either increase or set the weight to 0. In practice, the algorithm only loops over the columns currently stored and the newly arrived column, but the analysis implicitly considers the columns it dropped and has not seen yet. At the end of a full iteration, the columns are actually discarded. For the end of an iteration ($i = t$) we use the shortened notation $\widehat{\mathbf{Y}}_s = \widehat{\mathbf{Y}}_{t,s}$.

To bound $\lambda_{\max}(\widehat{\mathbf{Y}}_t)$, we will use Freedman's inequality, in particular its extensions for matrix martingales [20] from Proposition 3. Instead of working directly on $\widehat{\mathbf{Y}}_t$, we'll consider a different process that has access to information unaccessible to a realistic algorithm but can be analyzed more easily. Consider the process

$$\mathbf{Y}_{i,s} = \left(\left(\sum_{k=1}^{i} \left(1 - b_{k,s}\right) \psi_k \psi_k^\mathsf{T} \right) + \left(\sum_{k=i+1}^{t} \left(1 - b_{k,s-1}\right) \psi_k \psi_k^\mathsf{T} \right) \right) \mathbb{I}\left\{ \|\mathbf{Y}_{s-1}\| \leq \varepsilon \right\} + \mathbf{Y}_{s-1} \mathbb{I}\left\{ \|\mathbf{Y}_{s-1}\| \geq \varepsilon \right\}.$$

This sequence represents a variant of our algorithm that can detect if the previous iteration failed to construct an accurate approximation. When this failure happens, it stops the process and continues until the end without updating anything. It is clear to see that if any of the intermediate elements of the process violates the condition, the last element will violate it too. For the rest, $\mathbf{Y}_s$ behaves exactly like $\widehat{\mathbf{Y}}_s$. Therefore, we can write

$$\mathbb{P}\left(\lambda_{\max}(\widehat{\mathbf{Y}}_t) \geq \varepsilon\right) \leq \mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon\right).$$

If we denote with $\mathbf{W}_t$ the predictable quadratic variation of the process $\mathbf{Y}_t$, we can write

$$\begin{aligned}
&\mathbb{P}\left(\lambda_{\max}(\widehat{\mathbf{Y}}_t) \geq \varepsilon\right) \leq \mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon\right) \\
&= \mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon \cap \lambda_{\max}(\mathbf{W}_t) \leq \sigma^2\right) + \mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon \cap \lambda_{\max}(\mathbf{W}_t) > \sigma^2\right) \\
&\leq \mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon \cap \lambda_{\max}(\mathbf{W}_t) \leq \sigma^2\right) + \mathbb{P}\left(\lambda_{\max}(\mathbf{W}_t) > \sigma^2\right)
\end{aligned}$$

Where the last inequality derives from the definition of probability of an intersection of two events. For the first term, we will use Proposition 3, while for the second we can use Proposition 4. We will now show that our processes satisfy the conditions required to apply the propositions, and how to properly choose σ^2.

First, let's analyze the process $\mathbf{Y}_t$. We define the martingale difference as

$$\begin{aligned}
\mathbf{X}_{i,s} &= (b_{i,s-1} - b_{i,s})\,\psi_i\psi_i^{\mathsf{T}}\mathbb{I}\left\{\|\mathbf{Y}_{s-1}\| \leq \varepsilon\right\} + \mathbf{0}\mathbb{I}\left\{\|\mathbf{Y}_{s-1}\| \geq \varepsilon\right\} \\
&= (b_{i,s-1} - b_{i,s})\,\psi_i\psi_i^{\mathsf{T}}\mathbb{I}\left\{\|\mathbf{Y}_{s-1}\| \leq \varepsilon\right\}
\end{aligned}$$

First, we show that the martingale properties are satisfied. We begin by remarking that conditioned on all events up to time $s-1$, $\widetilde{p}_{i,s}$ is a fixed quantity. The only randomness across the iteration is the set of random variables $\{b_{i,s}\}$, which (again, conditioned on the previous iteration) are independent. Moreover, if the indicator function is not active, $\mathbf{Y}_s$ is deterministically equal to $\mathbf{Y}_{s-1}$ (the process is stopped), and the martingale requirement is satisfied. Therefore, for the reminder of the proof, we can safely assume $\|\mathbf{Y}_{s-1}\| \leq \varepsilon$. Define all the past filtration as $\mathcal{F}_{i,s} = \{\{\mathbf{X}_{k,s}\}_{k=0}^{i-1}, \mathbf{Y}_{s-1}\}$. Conditioned on $\mathcal{F}_{i,s}$, $b_{i,s-1}$ and $\widetilde{p}_{i,s}$ are constants. In this case,

$$\mathop{\mathbb{E}}_{b_{i,s}}\left[\mathbf{X}_{i,s} \mid \mathcal{F}_{i,s}\right] = \mathop{\mathbb{E}}_{b_{i,s}}\left[(b_{i,s-1} - b_{i,s})\,\psi_i\psi_i^{\mathsf{T}} \mid \mathcal{F}_{i,s}\right] = \left(b_{i,s-1} - \mathop{\mathbb{E}}_{b_{i,s}}\left[b_{i,s} \mid \mathcal{F}_{i,s}\right]\right)\psi_i\psi_i^{\mathsf{T}}$$

We should now compute the expected value of $b_{i,s}$. Without loss of generality, assume $b_{i,s-1} = l$. Therefore, the algorithm will continue to execute Bernoulli trials until $b_{i,s} \geq 1/(\widetilde{p}_{i,s}\overline{q})$. Notice that given $\mathbf{Y}_{s-1}$, this is a fixed quantity $l' > l$, that we will take as target. If any of the $l' - l$ trials fail, the variable will be set to 0. Therefore, its expected value is

$$\begin{aligned}
&\mathbb{E}\left[b_{i,s} \mid \mathcal{F}_{i,s}\right] = l'\mathbb{P}\left(\mathcal{B}\left(\frac{l'-1}{l'}\right) = 1 \cap \mathcal{B}\left(\frac{l'-2}{l'-1}\right) = 1 \cap \cdots \cap \mathcal{B}\left(\frac{l}{l+1}\right) = 1\right) \\
&= l'\mathbb{P}\left(\mathcal{B}\left(\frac{l'-1}{l'}\right) = 1\right)\mathbb{P}\left(\mathcal{B}\left(\frac{l'-2}{l'-1}\right) = 1\right)\mathbb{P}\left(\dots\right)\mathbb{P}\left(\mathcal{B}\left(\frac{l}{l+1}\right) = 1\right) \\
&= l'\frac{l'-1}{l'}\frac{l'-2}{l'-1}\cdots\frac{l}{l+1} = l = b_{i,s-1}
\end{aligned}$$

Then

$$\mathop{\mathbb{E}}_{b_{i,s}}\left[(b_{i,s-1} - b_{i,s})\,\psi_i\psi_i^{\mathsf{T}} \mid \mathcal{F}_{i,s}\right] = \left(b_{i,s-1} - b_{i,s}\frac{b_{i,s-1}}{b_{i,s}}\right)\psi_i\psi_i^{\mathsf{T}} = 0$$

This proves that our process is indeed a martingale. We can now continue with the properties necessary to apply Proposition 3. First we need to compute R in Proposition 3. In our modified process, all the way up to time $t-1$, we are guaranteed to have $\|\mathbf{Y}_{s-1}\| \leq \varepsilon$ (or the process is stopped). Under this condition, Alaoui and Mahoney [1, App. A, Lemma 1] show that it holds deterministically that

$$\frac{\tau_{i,s}}{\alpha\beta d_{\text{eff}}(\gamma)_s} = \frac{p_{i,s}}{\alpha\beta} \leq \widetilde{p}_{i,s} \leq p_{i,s} = \frac{\tau_{i,s}}{d_{\text{eff}}(\gamma)_s}$$

Therefore, we know that

$$\frac{\tau_{i,t}}{\alpha\beta d_{\text{eff}}(\gamma)_t} = \frac{p_{i,t}}{\alpha\beta} \leq \widetilde{p}_{i,t} \leq \widetilde{p}_{i,s}$$

From Subroutine Shrink-Expand, we know that the condition to try to increase $b_{i,s}$ is that $b_{i,s}\widetilde{p}_{i,s} \leq 1/\overline{q}$, let M_i the smallest integer such that

$$2\frac{\alpha\beta}{p_{i,t}\overline{q}} \geq \frac{\alpha\beta}{p_{i,t}\overline{q}} + 1 \geq M_i \geq \frac{\alpha\beta}{p_{i,t}\overline{q}} \geq \frac{1}{\widetilde{p}_{i,t}\overline{q}}.$$

from the condition, we know that the Algorithm will never try to increase $b_{i,s}$ above M_i, and that if a column reaches this quantity, we will surely keep it in the dictionary forever. Therefore

$$\begin{aligned}
\lambda_{\max}(\mathbf{X}_{i,s}) &= \lambda_{\max}\left(\left(b_{i,s-1} - b_{i,s}\right)\psi_i\psi_i^\mathsf{T}\right) \leq \lambda_{\max}\left(b_{i,s-1}\psi_i\psi_i^\mathsf{T}\right) \\
&\leq \lambda_{\max}\left(M_i\psi_i\psi_i^\mathsf{T}\right) \leq \lambda_{\max}\left(2\frac{\alpha\beta}{p_{i,t}\overline{q}}\psi_i\psi_i^\mathsf{T}\right) = \lambda_{\max}\left(\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\tau_{i,t}\overline{q}}\psi_i\psi_i^\mathsf{T}\right) \\
&\leq \lambda_{\max}\left(\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}\mathbf{I}\right) = \lambda_{\max}\left(\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}\mathbf{I}\right) \leq 2\alpha\beta d_{\text{eff}}(\gamma)_t/\overline{q} = R
\end{aligned}$$

We need now to analyze $\mathbf{W}_t$. Again, define all the past filtration as $\mathcal{F}_{i,s} = \{\{\mathbf{X}_{k,s}\}_{k=0}^{i-1}, \mathbf{Y}_{s-1}\}$. We have

$$\mathbf{W}_t = \sum_{s=1}^{t}\sum_{i=1}^{t}\mathbb{E}\left[\mathbf{X}_{i,s}^2 \;\middle|\; \mathcal{F}_{i,s}\right] = \sum_{s=1}^{t}\sum_{i=1}^{t}\mathbb{E}\left[\left(b_{i,s-1} - b_{i,s}\right)^2\psi_i\psi_i^\mathsf{T}\psi_i\psi_i^\mathsf{T} \;\middle|\; \mathcal{F}_{i,s}\right]$$

Again, without loss of generality, assume $b_{i,s} = l'$, $b_{i,s-1} = l$. We compute

$$\begin{aligned}
&\sum_{s=1}^{t}\mathbb{E}\left[\left(b_{i,s-1} - b_{i,s}\right)^2 \;\middle|\; \mathcal{F}_{i,s}\right] = \sum_{s=1}^{t} b_{i,s-1}^2 - 2b_{i,s-1}\,\mathbb{E}\left[b_{i,s} \mid \mathcal{F}_{i,s}\right] + \mathbb{E}\left[b_{i,s}^2 \;\middle|\; \mathcal{F}_{i,s}\right] \\
&= \sum_{s=1}^{t} b_{i,s-1}^2 - 2b_{i,s-1}^2 + \mathbb{E}\left[b_{i,s}^2 \;\middle|\; \mathcal{F}_{i,s}\right] = \sum_{s=1}^{t}\mathbb{E}\left[b_{i,s}^2 \;\middle|\; \mathcal{F}_{i,s}\right] - b_{i,s-1}^2
\end{aligned}$$

Now,

$$\begin{aligned}
&\mathbb{E}\left[b_{i,s}^2 \;\middle|\; \mathcal{F}_{i,s}\right] = {l'}^2\mathbb{P}\left(\mathcal{B}\left(\frac{l'-1}{l'}\right) = 1 \cap \mathcal{B}\left(\frac{l'-2}{l'-1}\right) = 1 \cap \cdots \cap \mathcal{B}\left(\frac{l}{l+1}\right) = 1\right) \\
&= {l'}^2\mathbb{P}\left(\mathcal{B}\left(\frac{l'-1}{l'}\right) = 1\right)\mathbb{P}\left(\mathcal{B}\left(\frac{l'-2}{l'-1}\right) = 1\right)\mathbb{P}\left(\dots\right)\mathbb{P}\left(\mathcal{B}\left(\frac{l}{l+1}\right) = 1\right) \\
&= {l'}^2\frac{l'-1}{l'}\frac{l'-2}{l'-1}\cdots\frac{l}{l+1} = l'l = b_{i,s}b_{i,s-1}
\end{aligned}$$

Let $B_i = \max_{s=1}^{t} b_{i,s}$ be the maximum achieved by $b_{i,s}$ before failing a Bernoulli trial and being set to zero forever, and let $d_i = \arg\max_{s=1}^{t} b_{i,s}$ be the timestep where this is achieved. We can see

$$\begin{aligned}
\mathbf{W}_t &= \sum_{i=1}^{t}\sum_{s=1}^{t} b_{i,s-1}(b_{i,s} - b_{i,s-1})\psi_i\psi_i^\mathsf{T}\psi_i\psi_i^\mathsf{T} = \sum_{i=1}^{t}\sum_{s=1}^{d_i} b_{i,s-1}(b_{i,s} - b_{i,s-1})\psi_i\psi_i^\mathsf{T}\psi_i\psi_i^\mathsf{T} \\
&= \sum_{i=1}^{t}\left(b_{i,d_i-1}b_{i,d_i} - \sum_{s=1}^{d_i-1} b_{i,s}(b_{i,s} - b_{i,s-1}) - b_{i,0}\right)\psi_i\psi_i^\mathsf{T}\psi_i\psi_i^\mathsf{T} \\
&\preceq \sum_{i=1}^{t} b_{i,d_i}b_{i,d_i}\psi_i\psi_i^\mathsf{T}\psi_i\psi_i^\mathsf{T} = \sum_{i=1}^{t} B_i^2\psi_i\psi_i^\mathsf{T}\psi_i\psi_i^\mathsf{T}
\end{aligned}$$

Where we eliminated the inner negative summation because each of its elements $b_{i,s}(b_{i,s} - b_{i,s-1})$ is surely positive or zero, and the overall summation is negative. We now have all the elements to apply the Freedman inequality, but we do not know which value of σ^2 will hold in high probability. Therefore, we will apply the Chernoff inequality to $\mathbf{W}_t$ itself. To begin, we note that the maximum number of trials that the algorithm will carry out on $b_{i,s}$ is bounded by $\frac{1}{\widetilde{p}_{i,t}\overline{q}} + 1$. Because $\widetilde{p}_{i,t}$ is not independent from $\widetilde{p}_{j,t}$, we have also that B_i is not independent from B_j. To simplify the analysis, we will consider a process that continue trying to increase $b_{i,s}$ until it reaches M_i. Define B_i' as the maximum achieved in this

augmented process. Clearly, $B'_i \geq B_i$, because we only give $b_{i,s}$ more possibilities to increase. But since M_i is a fixed quantity, B'_i is independent of B'_j. Therefore we can write

$$\mathbf{W}_t \preceq \sum_{i=1}^{t} B_i^2 \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} \preceq \sum_{i=1}^{t} B_i'^2 \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} = \sum_{i=1}^{t} \mathbf{V}_i$$

and analyze the sum of independent of matrices $\mathbf{V}_i$. Again, we know that the algorithm will never try to increase $b_{i,s}$ above M_i, and therefore $B_i \leq M_i$. We have

$$\begin{aligned}
\lambda_{\max}(\mathbf{V}_i) &= \lambda_{\max}\left(B_i'^2 \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T}\right) \leq \lambda_{\max}\left(M_i^2 \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T}\right) \\
&\leq \lambda_{\max}\left(\left(\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\tau_{i,t}\overline{q}}\right)^2 \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T}\right) \\
&\leq \lambda_{\max}\left(\left(\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}\right)^2 \mathbf{I}\right) \leq 4\left(\frac{\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}\right)^2 = R
\end{aligned}$$

To compute μ we derive

$$\begin{aligned}
\mathbb{E}[\mathbf{V}_s] &= \mathop{\mathbb{E}}_{1:M_i}\left[\sum_{k=1}^{M_i} k^2 \mathbb{I}\{b_{i,k+1} = 0 \cap b_{i,k} \neq 0\} \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T}\right] \\
&= \sum_{k=1}^{M_i} \mathop{\mathbb{E}}_{1:k}\left[k^2 \mathbb{I}\{b_{i,k+1} = 0 \cap b_{i,k} \neq 0\} \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T}\right] = \sum_{k=1}^{M_i} k^2 \mathbb{P}(b_{i,k+1} = 0)\mathbb{P}(b_{i,k} \neq 0) \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} \\
&= \sum_{k=1}^{M_i} k^2 \left(1 - \frac{k}{k+1}\right) \frac{1}{k} \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} = \sum_{k=1}^{M_i} k \left(\frac{k+1-k}{k+1}\right) \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} = \sum_{k=1}^{M_i} \frac{k}{k+1} \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} \\
&\preceq \sum_{k=1}^{M_i} \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} = M_i \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} \preceq \frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\tau_{i,t}\overline{q}} \psi_i \psi_i^\mathsf{T} \psi_i \psi_i^\mathsf{T} \preceq \frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}} \psi_i \psi_i^\mathsf{T}
\end{aligned}$$

Therefore

$$\mu = \left\|\sum_{i=1}^{t} \mathbb{E}[\mathbf{V}_i]\right\|_2 \leq \left\|\sum_j \frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}} \psi_i \psi_i^\mathsf{T}\right\|_2 = \left\|\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}} \mathbf{\Psi}\mathbf{\Psi}^\mathsf{T}\right\|_2 \leq \frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}$$

We can now apply Proposition 4 with μ, $R = \mu^2$ and $\delta = 2$.

$$\begin{aligned}
&\mathbb{P}\left(\lambda_{\max}\left(\sum_j \mathbf{X}_j\right) \geq 3\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}\right) \leq \mathbb{P}\left(\lambda_{\max}\left(\sum_j \mathbf{X}_j\right) \geq (1+\delta)\mu\right) \\
&\leq t\left(\frac{e^2}{27}\right)^{1/\mu} \leq t\exp\left\{-\frac{1}{\mu}(\log(27) - 2)\right\} \leq t\exp\left\{-\frac{\overline{q}}{2\alpha\beta d_{\text{eff}}(\gamma)_t}\right\}.
\end{aligned}$$

Therefore with high probability we have

$$\lambda_{\max}(\mathbf{W}_t) \leq \lambda_{\max}\left(\sum_{i=1}^{t} \mathbf{V}_i\right) \leq 3\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}} = \sigma^2$$

Plugging this in the Freedman bound

$$\begin{aligned}
&\mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon \cap \lambda_{\max}(\mathbf{W}_t) \leq 3\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}\right) \\
&\leq t \cdot \exp\left\{-\frac{-\varepsilon^2/2}{(3+\varepsilon/3)\frac{2\alpha\beta d_{\text{eff}}(\gamma)_t}{\overline{q}}}\right\} = t \cdot \exp\left\{-\frac{\varepsilon^2/2}{(3+\varepsilon/3)}\frac{\overline{q}}{2\alpha\beta d_{\text{eff}}(\gamma)_t}\right\}.
\end{aligned}$$

Therefore, by carefully choosing $\overline{q}$ we have that

$$\begin{aligned}
&\mathbb{P}\left(\lambda_{\max}(\widehat{\mathbf{Y}}_t) \geq \varepsilon\right) \leq \mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon\right) \\
&\leq \mathbb{P}\left(\lambda_{\max}(\mathbf{Y}_t) \geq \varepsilon \cap \lambda_{\max}(\mathbf{W}_t) \leq \sigma^2\right) + \mathbb{P}\left(\lambda_{\max}(\mathbf{W}_t) > \sigma^2\right) \leq \frac{\delta}{4t} + \frac{\delta}{4t} \leq \frac{\delta}{2t}
\end{aligned}$$

Space complexity We must now separately bound the number of columns present in the dictionary at each time step. This is equivalent, at time t, to counting how many $b_{i,t}$ are different than 0. Again, from the terminating condition in the algorithm, we know that

$$\mathbb{P}(b_{i,t} \neq 0) = 1/b_{i,t} \leq \overline{q}\widetilde{p}_{i,t} \leq \overline{q}p_{i,t}$$

Let $z_{i,t} = \mathbb{I}\{b_{i,t} \neq 0\}$ be the random variable that indicates whether we column i survived until time t or not. Notice the trial of all columns are independent. By definition $z_{i,t}$ are Bernoulli random variables and their probability parameter is upper bounded by $\overline{p}_{i,t} = \min\{\overline{q}p_{i,t}, 1\}$. We can rewrite the total number of columns kept as

$$Q_t = \sum_{i=1}^{t} z_{i,t}$$

First, it is easy to see that $\mathbb{E}[\sum_{i=1}^{t} z_i] \leq \overline{q}$. Using Hoeffding bound we can show

$$\begin{aligned}
&\mathbb{P}\left(\sum_{i=1}^{t} z_i \geq g\overline{q}\right) = \inf_{\theta} \mathbb{P}\left(e^{\sum_{i=1}^{t}\theta z_i} \geq e^{\theta g\overline{q}}\right) \\
&\leq \inf_{\theta} \frac{\mathbb{E}\left[e^{\sum_{i=1}^{t}\theta z_i}\right]}{e^{\theta g\overline{q}}} = \inf_{\theta} \frac{\mathbb{E}\left[\prod_{i=1}^{t} e^{\theta z_i}\right]}{e^{\theta g\overline{q}}} = \inf_{\theta} \frac{\prod_{i=1}^{t}\mathbb{E}\left[e^{\theta z_i}\right]}{e^{\theta g\overline{q}}} \\
&= \inf_{\theta} \frac{\prod_{i=1}^{t}(\overline{p}_{i,t}e^{\theta} + (1-\overline{p}_{i,t}))}{e^{\theta g\overline{q}}} = \inf_{\theta} \frac{\prod_{i=1}^{t}(1+\overline{p}_{i,t}(e^{\theta}-1))}{e^{\theta g\overline{q}}} \leq \inf_{\theta} \frac{\prod_{i=1}^{t} e^{\overline{p}_{i,t}(e^{\theta}-1)}}{e^{\theta g\overline{q}}} \\
&\leq \inf_{\theta} \frac{e^{\overline{q}(e^{\theta}-1)}}{e^{\theta g\overline{q}}} = \inf_{\theta} e^{(\overline{q}e^{\theta}-\overline{q}-\theta g\overline{q})},
\end{aligned}$$

where we use the fact that $\sum_{i=1}^{t} p_{i,t} \leq 1$. The choice of θ minimizing the previous expression is obtained as

$$\frac{d}{d\theta} e^{(\overline{q}e^{\theta}-\overline{q}-\theta g\overline{q})} = e^{(\overline{q}e^{\theta}-\overline{q}-\theta g\overline{q})}(\overline{q}e^{\theta} - g\overline{q}) = 0,$$

and thus $\theta = \log(g)$. Finally

$$\mathbb{P}\left(\sum_{i=1}^{t} z_i \geq g\overline{q}\right) \leq \inf_{\theta} e^{\overline{q}(e^{\theta}-1-\theta g)} = e^{\overline{q}(g-1-g\log g)} \leq e^{-\overline{q}g(\log g - 1)}.$$

Taking $g = e^2$, we have $\log(g) = 2$ and that gives us

$$\begin{aligned}
\mathbb{P}\left(\sum_{i=1}^{t} z_i \geq e^2\overline{q}\right) &\leq \exp\left\{-e^2\overline{q}(2-1)\right\} \\
&\leq \exp\left\{-e^2 \frac{28\alpha\beta d_{\text{eff}}(\gamma)_t}{\varepsilon^2}\log\left(\frac{4t}{\delta}\right)\right\} = \left(\frac{\delta}{4t}\right)^{\left(e^2\frac{28\alpha\beta d_{\text{eff}}(\gamma)_t}{\varepsilon^2}\right)} \leq \frac{\delta}{2t}.
\end{aligned}$$

Therefore

$$\mathbb{P}\left(\left\|\boldsymbol{\Psi}\boldsymbol{\Psi}^{\mathsf{T}} - \boldsymbol{\Psi}\mathbf{S}_t\mathbf{S}_t^{\mathsf{T}}\boldsymbol{\Psi}^{\mathsf{T}}\right\|_2 \geq \varepsilon \cup Q_t \geq 8\overline{q}\right) \leq \frac{\delta}{2t} + \frac{\delta}{2t} \leq \frac{\delta}{t},$$

and this concludes the proof.

□

Proof or Corollary 1. To simplify the proof, we first introduce the quantities $\gamma = t\gamma'$ and $\mu = t\mu'$ and $\gamma/\mu = \gamma'/\mu'$. We begin by decomposing the generalization error in a bias and variance part,

$$\begin{aligned}
&\mathcal{R}(\widetilde{\mathbf{w}}_t) = \mathrm{E}_{\psi}\|\widetilde{\mathbf{K}}_t(\widetilde{\mathbf{K}}_t + t\mu' I)^{-1}(\mathbf{f}^* + \sigma^2\xi) - \mathbf{f}^*\|_2^2 \\
&= \|(\widetilde{\mathbf{K}}_t(\widetilde{\mathbf{K}}_t + t\mu'\mathbf{I})^{-1} - \mathbf{I})\mathbf{f}^*\|_2^2 + \sigma^2\mathrm{E}_{\psi}\|\widetilde{\mathbf{K}}_t(\widetilde{\mathbf{K}}_t + t\mu'\mathbf{I})^{-1}\xi\|_2^2 \\
&= t^2\mu'^2\|(\widetilde{\mathbf{K}}_t + t\mu'\mathbf{I})^{-1}\mathbf{f}^*\|_2^2 + \sigma^2\,\mathrm{Tr}(\widetilde{\mathbf{K}}_t^2(\widetilde{\mathbf{K}}_t + t\mu'\mathbf{I})^{-2}) \\
&:= \mathrm{bias}(\widetilde{\mathbf{K}}_t)^2 + \mathrm{variance}(\widetilde{\mathbf{K}}_t)
\end{aligned}$$

From the proof of Alaoui and Mahoney [1, App. A, Lemma 2], we have

$$\begin{aligned}\|(\widetilde{\mathbf{K}}_t + t\mu'\mathbf{I})^{-1}\mathbf{f}^*\|_2 \leq& \|(\mathbf{K}_t + t\mu'\mathbf{I})^{-1}\mathbf{f}^*\|_2 \left(1 + \frac{t\gamma'}{1-\varepsilon}\|(\widetilde{\mathbf{K}}_t + t\mu'\mathbf{I})^{-1}\|_2\right)\\ \leq& \|(\mathbf{K}_t + t\mu'\mathbf{I})^{-1}\mathbf{f}^*\|_2 \left(1 + \frac{\gamma'/\mu'}{1-\varepsilon}\right)\end{aligned}$$

Therefore

$$\text{bias}(\widetilde{\mathbf{K}}_t)^2 \leq \left(1 + \frac{\gamma'/\mu'}{1-\varepsilon}\right)^2 \text{bias}(\mathbf{K}_t)^2$$

It is easy to see that the variance decreases if we use $\widetilde{\mathbf{K}}_t$ instead of $\mathbf{K}_t$. We can rewrite the variance as

$$\text{variance}(\mathbf{K}_t) = \text{Tr}(\mathbf{K}_t^2(\mathbf{K}_t + t\mu\mathbf{I})^{-2}) = \sum_{i=1}^{t} \frac{\lambda_i^2}{(\lambda_i + t\mu')^2},$$

where λ_i are the eigenvalues of the kernel matrix and it shows that the variace is strictly increasing in λ_i. Because $\widetilde{\mathbf{K}}_t \preceq \mathbf{K}_t$, the same ordering applies to each eigenvalue of the two matrices, and therefore

$$\text{variance}(\widetilde{\mathbf{K}}_t) \leq \text{variance}(\mathbf{K}_t).$$

Putting it all together

$$\begin{aligned}\mathcal{R}(\widetilde{\mathbf{w}}_t) =& \text{bias}(\widetilde{\mathbf{K}}_t)^2 + \text{variance}(\widetilde{\mathbf{K}}_t)\\ \leq& \left(1 + \frac{\gamma'/\mu'}{1-\varepsilon}\right)^2 \text{bias}(\mathbf{K}_t)^2 + \text{variance}(\mathbf{K}_t)\\ \leq& \left(1 + \frac{\gamma'/\mu'}{1-\varepsilon}\right)^2 (\text{bias}(\mathbf{K}_t)^2 + \text{variance}(\mathbf{K}_t)) = \left(1 + \frac{\gamma'/\mu'}{1-\varepsilon}\right)^2 \mathcal{R}(\widehat{\mathbf{w}}_t)\\ =& \left(1 + \frac{\gamma/\mu}{1-\varepsilon}\right)^2 \mathcal{R}(\widehat{\mathbf{w}}_t)\end{aligned}$$

□

C Extended proofs of Section 4

Proof of Lemma 2. Subtracting $\mathbf{K}_{t+1} - \overline{\mathbf{K}}_{t+1}$ and recalling the block matrix multiplication formula we have

$$\left[\begin{array}{c|c} \mathbf{x}_{t+1}^\mathsf{T} & x_{t+1}\end{array}\right] \left[\begin{array}{c|c} \mathbf{K}_t - \widetilde{\mathbf{K}}_t & \mathbf{0} \\ \hline \mathbf{0}^\mathsf{T} & 0\end{array}\right] \left[\begin{array}{c} \mathbf{x}_{t+1} \\ \hline x_{t+1}\end{array}\right] = \mathbf{x}_{t+1}^\mathsf{T}(\mathbf{K}_t - \overline{\mathbf{K}})\mathbf{x}_{t+1} \leq \frac{\gamma}{1-\varepsilon}\mathbf{x}_{t+1}^\mathsf{T}\mathbf{x}_{t+1}$$

Therefore, $\overline{\mathbf{K}}_{t+1}$ satisfies (11). For the remainder of the proof we drop the dependency on time $t+1$, and simply write $\mathbf{K}$ and $\overline{\mathbf{K}}$. Let $\eta = \frac{2-\varepsilon}{1-\varepsilon}$, from Proposition 1, we derive

$$(\mathbf{K} + \gamma\mathbf{I})^{-1} \succeq (\overline{\mathbf{K}} + \eta\gamma\mathbf{I})^{-1} \succeq (\mathbf{K} + \eta\gamma\mathbf{I})^{-1} \succeq \frac{1}{\eta}(\mathbf{K} + \gamma\mathbf{I})^{-1}$$

We need to prove something along the lines of

$$\begin{aligned}\tau_i =& \mathbf{k}_i^\mathsf{T}(\mathbf{K} + \gamma\mathbf{I})^{-1}\mathbf{e}_i = \mathbf{e}_i^\mathsf{T}\mathbf{K}(\mathbf{K} + \gamma\mathbf{I})^{-1}\mathbf{e}_i = \mathbf{e}_i^\mathsf{T}\mathbf{K}^{1/2}(\mathbf{K} + \gamma\mathbf{I})^{-1}\mathbf{K}^{1/2}\mathbf{e}_i\\ \geq& \mathbf{e}_i^\mathsf{T}\mathbf{K}^{1/2}\left(\overline{\mathbf{K}} + \eta\gamma\mathbf{I}\right)^{-1}\mathbf{K}^{1/2}\mathbf{e}_i \geq \frac{1}{\eta}\mathbf{e}_i^\mathsf{T}\mathbf{K}^{1/2}(\mathbf{K} + \gamma\mathbf{I})^{-1}\mathbf{K}^{1/2}\mathbf{e}_i = \frac{1}{\eta}\tau_i\end{aligned}$$

where the middle line would be our estimator. The problem is that we do not have access to $\mathbf{K}^{1/2}$ (it would take too much time and space to compute), and we do not want our bound to depend on the smallest eigenvalue.

We can proceed as follows. Differently from Alaoui and Mahoney [1] we will only look to approximate leverage scores for columns in $\mathbf{S}$, or in other words only entries strictly on the diagonal of $\mathbf{K}(\mathbf{K}+\gamma)^{-1}$, and only for columns that we fully store.

We begin by noting

$$\tau_i = \mathbf{k}_i(\mathbf{K}+\gamma\mathbf{I})^{-1}\mathbf{e}_i = \frac{1}{\gamma}\mathbf{e}_i\mathbf{K}(\mathbf{K}+\gamma\mathbf{I})^{-1}(\gamma\mathbf{I})\mathbf{e}_i = \frac{1}{\gamma}\mathbf{e}_i\mathbf{K}(\mathbf{K}+\gamma\mathbf{I})^{-1}(\mathbf{K}-\mathbf{K}+\gamma\mathbf{I})\mathbf{e}_i$$
$$\frac{1}{\gamma}\mathbf{e}_i\mathbf{K}(\mathbf{K}+\gamma\mathbf{I})^{-1}(\mathbf{K}+\gamma\mathbf{I})\mathbf{e}_i - \frac{1}{\gamma}\mathbf{e}_i\mathbf{K}(\mathbf{K}+\gamma\mathbf{I})^{-1}\mathbf{K}\mathbf{e}_i = \frac{1}{\gamma}(k_{i,i} - \mathbf{k}_i(\mathbf{K}+\gamma\mathbf{I})^{-1}\mathbf{k}_i)$$

We can easily see that

$$\begin{aligned}\tau_i =& \frac{1}{\gamma}(k_{i,i} - \mathbf{k}_i(\mathbf{K}+\gamma\mathbf{I})^{-1}\mathbf{k}_i)\\ &\leq \frac{1}{\gamma}\left(k_{i,i} - \mathbf{k}_i\left(\overline{\mathbf{K}}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{k}_i\right) \leq \frac{1}{\gamma}\left(k_{i,i} - \mathbf{k}_i\left(\mathbf{K}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{k}_i\right)\end{aligned}$$

Now

$$\begin{aligned}&k_{i,i} - \mathbf{k}_i(\mathbf{K}+\eta\gamma\mathbf{I})^{-1}\mathbf{k}_i = \mathbf{e}_i^\mathsf{T}\mathbf{K}\mathbf{e}_i - \mathbf{e}_i^\mathsf{T}\mathbf{K}\left(\mathbf{K}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{K}\mathbf{e}_i\\ &= \mathbf{e}_i^\mathsf{T}\mathbf{K}\left(\mathbf{I} - \mathbf{K}\left(\mathbf{K}+\eta\gamma\mathbf{I}\right)^{-1}\right)\mathbf{e}_i = \eta\gamma\mathbf{e}_i^\mathsf{T}\mathbf{K}\left(\mathbf{K}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{e}_i\\ &\leq \eta\gamma\mathbf{e}_i^\mathsf{T}\mathbf{K}(\mathbf{K}+\gamma\mathbf{I})^{-1}\mathbf{e}_i = \eta\gamma\tau\end{aligned}$$

Putting it all together

$$\tau \leq \frac{1}{\gamma}\left(k_{i,i} - \mathbf{k}_i\left(\overline{\mathbf{K}}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{k}_i\right) \leq \frac{1}{\gamma}\left(k_{i,i} - \mathbf{k}_i\left(\mathbf{K}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{k}_i\right) \leq \eta\frac{\gamma}{\gamma}\tau$$

□

Proof of Lemma 3. This proof proceeds in two steps. First, we will find upper and lower bounds for the term reported in Equation 16. Then we will use induction, and the fact that we can compute $d_{\text{eff}}(\gamma)_0$ exactly to prove the claim. Let $\eta = \frac{2-\varepsilon}{1-\varepsilon}$ We begin by reminding that as a consequence of Proposition 1, and of properties of nonsingular PSD matrices, we have

$$(\mathbf{K}+\gamma\mathbf{I})^{-1} \succeq \left(\widetilde{\mathbf{K}}_t + \eta\gamma_\varepsilon\mathbf{I}\right)^{-1} \succeq (\mathbf{K}+\eta\gamma\mathbf{I})^{-1} \succeq \frac{1}{\eta}(\mathbf{K}+\gamma\mathbf{I})^{-1}$$

We remind that $\mathbf{K}_t = \mathbf{\Phi}_t^\mathsf{T}\mathbf{\Phi}_t = \mathbf{U}\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}$ and $\overline{\mathbf{k}}_{t+1} = \mathbf{\Phi}_t^\mathsf{T}\phi$ with $\mathbf{\Phi}_t = \mathbf{V}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}$. Similarly, we introduce $\widetilde{\mathbf{K}}_t = \widetilde{\mathbf{U}}\widetilde{\mathbf{\Sigma}}^\mathsf{T}\widetilde{\mathbf{\Sigma}}\widetilde{\mathbf{U}}^\mathsf{T}$ we have now

$$(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\gamma\mathbf{I})^{-1} \succeq \mathbf{U}^\mathsf{T}\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{\Sigma}}^\mathsf{T}\widetilde{\mathbf{\Sigma}}+\eta\gamma\mathbf{I}\right)^{-1}\widetilde{\mathbf{U}}^\mathsf{T}\mathbf{U} \succeq \left(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\eta\gamma\mathbf{I}\right)^{-1}$$

We need a bound on

$$\widetilde{\Delta}_t = \frac{\left(k_{t+1} - \overline{\mathbf{k}}_{t+1}^\mathsf{T}\left(\widetilde{\mathbf{K}}_t^\gamma + \eta\gamma\mathbf{I}\right)^{-1}\overline{\mathbf{k}}_{t+1} - \frac{(1-\varepsilon)^2}{4}\gamma\overline{\mathbf{k}}_{t+1}^\mathsf{T}(\widetilde{\mathbf{K}}_t^\gamma+\gamma\mathbf{I})^{-2}\overline{\mathbf{k}}_{t+1}\right)}{\frac{1}{\eta}(k_{t+1}+\gamma - \overline{\mathbf{k}}_{t+1}^\mathsf{T}\left(\widetilde{\mathbf{K}}_t^\gamma+\eta\gamma\mathbf{I}\right)^{-1}\overline{\mathbf{k}}_{t+1})}$$

We will first upper and lower bound the denumerator

$$\begin{aligned}&k_{t+1}+\gamma - \overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t+\gamma\mathbf{I})^{-1}\overline{\mathbf{k}}_{t+1}\\ &\leq k_{t+1}+\gamma - \overline{\mathbf{k}}_{t+1}^\mathsf{T}\left(\widetilde{\mathbf{K}}_t^\gamma+\eta\gamma\mathbf{I}\right)^{-1}\overline{\mathbf{k}}_{t+1}\\ &= \gamma + \phi^\mathsf{T}\mathbf{V}\mathbf{V}^\mathsf{T}\phi - \phi^\mathsf{T}\mathbf{V}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}\widetilde{\mathbf{U}}\left(\widetilde{\mathbf{\Sigma}}^\mathsf{T}\widetilde{\mathbf{\Sigma}}+\eta\gamma\mathbf{I}\right)^{-1}\widetilde{\mathbf{U}}^\mathsf{T}\mathbf{U}\mathbf{\Sigma}^\mathsf{T}\mathbf{V}^\mathsf{T}\phi\\ &\leq \gamma + \phi^\mathsf{T}\mathbf{V}\mathbf{V}^\mathsf{T}\phi - \phi^\mathsf{T}\mathbf{V}\mathbf{\Sigma}\left(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{\Sigma}^\mathsf{T}\mathbf{V}^\mathsf{T}\phi\\ &= \gamma + \phi^\mathsf{T}\mathbf{V}\left(\mathbf{I} - \mathbf{\Sigma}\left(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{\Sigma}^\mathsf{T}\right)\mathbf{V}^\mathsf{T}\phi\end{aligned}$$

The last expression corresponds to a block diagonal matrix, where for all $i > t$, only the diagonal remains, with 1 on the diagonal, that we can easily upper bound with 2, because $1 < 2$ for all choiches of 1 and 2. We want to study the i-th entry in the diagonal matrix

$$1 - \frac{\sigma^2}{\sigma^2 + \eta\gamma} = \eta\frac{\gamma}{\sigma^2 + \eta\gamma} \leq \eta\frac{\gamma}{\sigma_2 + \gamma} = \eta\left(1 - \frac{\sigma^2}{\sigma^2 + \gamma}\right)$$

Thus,

$$\begin{aligned}
&\gamma + \phi^\mathsf{T}\mathbf{V}\left(\mathbf{I} - \mathbf{\Sigma}\left(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma} + \eta\gamma\mathbf{I}\right)^{-1}\mathbf{\Sigma}^\mathsf{T}\right)\mathbf{V}^\mathsf{T}\phi \\
&\leq \gamma + \eta\phi^\mathsf{T}\mathbf{V}(\mathbf{I} - \mathbf{\Sigma}(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma} + \gamma\mathbf{I})^{-1}\mathbf{\Sigma}^\mathsf{T})\mathbf{V}^\mathsf{T}\phi \\
&\leq \eta(\gamma + \phi^\mathsf{T}\mathbf{V}(\mathbf{I} - \mathbf{\Sigma}(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma} + \gamma\mathbf{I})^{-1}\mathbf{\Sigma}^\mathsf{T})\mathbf{V}^\mathsf{T}\phi) \\
&= \eta(k_{t+1} + \gamma - \overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t + \gamma\mathbf{I})^{-1}\overline{\mathbf{k}}_{t+1})
\end{aligned}$$

We should now bound the numerator. We will also need bounds on the squares of the inequalities we used this far. In particular we will begin with the one we have for free,

$$(\mathbf{K}_t - \widetilde{\mathbf{K}}_t)^2 \preceq \frac{\gamma^2}{(1-\varepsilon)^2}\mathbf{I},$$

because $\mathbf{I}$ commutes with everything. Given PD matrices $\mathbf{A}$ and $\mathbf{B}$ we have $\mathbf{A}^2 \geq \mathbf{B}^2$ if and only if the largest singular value of $\mathbf{A}^{-1}\mathbf{B}$ is smaller or equal than 1. We begin with

$$(\mathbf{K}_t + \gamma\mathbf{I})^2 \preceq \left(\frac{2}{1-\varepsilon}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\right)^2.$$

The singular values of matrix $\mathbf{A}$ are the square root of the eigenvalues of $\mathbf{A}^\mathsf{T}\mathbf{A}$ or $\mathbf{A}\mathbf{A}^\mathsf{T}$. They can be also defined as $\|\mathbf{A}\|_2$. In our case, we want to show

$$\left\|(\mathbf{K}_t + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 \leq 1.$$

We have

$$\begin{aligned}
&\left\|(\mathbf{K}_t + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 = \left\|(\mathbf{K}_t - \widetilde{\mathbf{K}}_t + \widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 \\
&\leq \left\|(\mathbf{K}_t - \widetilde{\mathbf{K}})\left(\frac{2}{1-\varepsilon}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 + \left\|(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2
\end{aligned}$$

Because $(\mathbf{K}_t - \widetilde{\mathbf{K}})^2 \preceq \gamma^2/(1-\varepsilon)^2\mathbf{I}$ holds for the squares, we can use it on the first term and obtain

$$\left\|(\mathbf{K}_t + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 \leq \frac{(1-\varepsilon)}{2}\left(\max_i \frac{\gamma}{(1-\varepsilon)(\widetilde{\lambda}_i + \gamma)} + \max_i \frac{\widetilde{\lambda}_i + \gamma}{\widetilde{\lambda}_i + \gamma}\right) \leq \frac{1}{2}2$$

Similarly for

$$(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})^2 \preceq \left(\frac{2}{1-\varepsilon}(\mathbf{K}_t + \gamma\mathbf{I})\right)^2$$

We have

$$\begin{aligned}
&\left\|(\widetilde{\mathbf{K}}_t + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\mathbf{K}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 = \left\|(\widetilde{\mathbf{K}}_t - \mathbf{K}_t + \mathbf{K} + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\mathbf{K}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 \\
&\leq \left\|(\widetilde{\mathbf{K}}_t - \mathbf{K}_t)\left(\frac{2}{1-\varepsilon}(\mathbf{K}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2 + \left\|(\mathbf{K} + \gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\mathbf{K}_t + \gamma\mathbf{I})\right)^{-1}\right\|_2
\end{aligned}$$

And

$$\left\|(\widetilde{\mathbf{K}}_t+\gamma\mathbf{I})\left(\frac{2}{1-\varepsilon}(\mathbf{K}_t+\gamma\mathbf{I})\right)^{-1}\right\|_2\leq\frac{1-\varepsilon}{2}\left(\max_i\frac{\gamma}{(1-\varepsilon)(\lambda_i+\gamma)}+\max_i\frac{\lambda_i+\gamma}{\lambda_i+\gamma}\right)\leq\frac{1}{2}2$$

Therefore

$$\frac{(1-\varepsilon)^4}{16}(\mathbf{K}_t+\gamma\mathbf{I})^{-2}\preceq\frac{(1-\varepsilon)^2}{4}(\widetilde{\mathbf{K}}_t+\gamma\mathbf{I})^{-2}\preceq(\mathbf{K}_t+\gamma\mathbf{I})^{-2}$$

Similarly to the denominator, we derive

$$\begin{aligned}
&k_{t+1}-\overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t+\gamma\mathbf{I})^{-1}\overline{\mathbf{k}}_{t+1}-\gamma\overline{\mathbf{k}}_{t+1}^\mathsf{T}(\mathbf{K}_t+\gamma\mathbf{I})^{-2}\overline{\mathbf{k}}_{t+1}\\
\leq&k_{t+1}-\overline{\mathbf{k}}_{t+1}^\mathsf{T}\left(\widetilde{\mathbf{K}}_t^\gamma+\eta\gamma\mathbf{I}\right)^{-1}\overline{\mathbf{k}}_{t+1}-\frac{(1-\varepsilon)^2}{4}\gamma\overline{\mathbf{k}}_{t+1}^\mathsf{T}(\widetilde{\mathbf{K}}_t^\gamma+\gamma\mathbf{I})^{-2}\overline{\mathbf{k}}_{t+1}\\
=&\phi^\mathsf{T}\mathbf{V}\mathbf{V}^\mathsf{T}\phi-\phi^\mathsf{T}\mathbf{V}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}\widetilde{\mathbf{U}}(\widetilde{\mathbf{\Sigma}}^\mathsf{T}\widetilde{\mathbf{\Sigma}}+\eta\gamma\mathbf{I})^{-1}\widetilde{\mathbf{U}}^\mathsf{T}\mathbf{U}\mathbf{\Sigma}^\mathsf{T}\mathbf{V}^\mathsf{T}\phi-\frac{(1-\varepsilon)^2}{4}\gamma\phi^\mathsf{T}\mathbf{V}\mathbf{\Sigma}\mathbf{U}^\mathsf{T}\widetilde{\mathbf{U}}(\widetilde{\mathbf{\Sigma}}^\mathsf{T}\widetilde{\mathbf{\Sigma}}+\gamma\mathbf{I})^{-2}\widetilde{\mathbf{U}}^\mathsf{T}\mathbf{U}\mathbf{\Sigma}^\mathsf{T}\mathbf{V}^\mathsf{T}\phi\\
\leq&\phi^\mathsf{T}\mathbf{V}\mathbf{V}^\mathsf{T}\phi-\phi^\mathsf{T}\mathbf{V}\mathbf{\Sigma}\left(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{\Sigma}^\mathsf{T}\mathbf{V}^\mathsf{T}\phi-\frac{(1-\varepsilon)^4}{16}\gamma\phi^\mathsf{T}\mathbf{V}\mathbf{\Sigma}(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\gamma\mathbf{I})^{-2}\mathbf{\Sigma}^\mathsf{T}\mathbf{V}^\mathsf{T}\phi\\
=&\phi^\mathsf{T}\mathbf{V}\left(\mathbf{I}-\mathbf{\Sigma}\left(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\eta\gamma\mathbf{I}\right)^{-1}\mathbf{\Sigma}^\mathsf{T}-\frac{(1-\varepsilon)^4}{16}\gamma\mathbf{\Sigma}(\mathbf{\Sigma}^\mathsf{T}\mathbf{\Sigma}+\gamma\mathbf{I})^{-2}\mathbf{\Sigma}^\mathsf{T}\right)\mathbf{V}^\mathsf{T}\phi
\end{aligned}$$

Again, we study the object

$$\begin{aligned}
&1-\frac{\sigma^2}{\sigma^2+\eta\gamma}-\frac{(1-\varepsilon)^4}{16}\frac{\gamma\sigma^2}{(\sigma^2+\gamma)^2}\leq1-\frac{\sigma^2}{\sigma^2+\eta\gamma}-\frac{(1-\varepsilon)^4}{16}\frac{\gamma\sigma^2}{(\sigma^2+\eta\gamma)(\sigma^2+\gamma)}\\
&=\frac{\sigma^4+(1+\eta)\gamma\sigma^2+\eta\gamma^2-\sigma^4-\gamma\sigma^2-((1-\varepsilon)^4/16)\gamma\sigma^2}{(\sigma^2+\eta\gamma)(\sigma^2+\gamma)}\\
&\leq\frac{\eta\gamma\sigma^2+\eta\gamma^2}{(\sigma^2+\gamma)^2}\leq\eta\left(1+\frac{\sigma^2}{\gamma}\right)\frac{\gamma^2}{(\sigma^2+\gamma)^2}\leq\eta\left(1+\frac{\lambda_{\max}}{\gamma}\right)\frac{\gamma^2}{(\sigma^2+\gamma)^2}
\end{aligned}$$

Therefore

$$\Delta_t\leq\widetilde{\Delta}_t\leq\eta^2\left(1+\frac{\lambda_{\max}}{\gamma}\right)\Delta_t$$

Now assume the inductive hypotheses that $\widetilde{d}_{\text{eff}}(\gamma)_t$ is β-approximate holds, with $\beta=\eta^2\left(1+\frac{\lambda_{\max}}{\gamma}\right)$, and we can see that

$$\begin{aligned}
\widetilde{d}_{\text{eff}}(\gamma)_{t+1}&=\widetilde{d}_{\text{eff}}(\gamma)_t+\widetilde{\Delta}_t\leq\beta(d_{\text{eff}}(\gamma)_t+\Delta_t)=\beta d_{\text{eff}}(\gamma)_{t+1}\\
\widetilde{d}_{\text{eff}}(\gamma)_{t+1}&=\widetilde{d}_{\text{eff}}(\gamma)_t+\widetilde{\Delta}_t\geq d_{\text{eff}}(\gamma)_t+\Delta_t=d_{\text{eff}}(\gamma)_{t+1}
\end{aligned}$$

Therefore $\widetilde{d}_{\text{eff}}(\gamma)_{t+1}$ is also a β-approximation. Because at the first iteration we can compute $\widetilde{d}_{\text{eff}}(\gamma)_0$ exactly, we can prove Lemma 3 by induction. □